

Владимирский государственный университет

Д. А. Якубович Е. С. Еропова

ИЗДАТЕЛЬСКАЯ СИСТЕМА LaTeX

Учебное пособие

Владимир 2019

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Д. А. Якубович Е. С. Еропова

ИЗДАТЕЛЬСКАЯ СИСТЕМА LaTeX

Учебное пособие

Электронное издание



Владимир 2019

© ВлГУ, 2019

© Якубович Д. А., Еропова Е. С., 2019

ISBN 978-5-9984-0956-1

УДК 004.915
ББК 32.972

Рецензенты:

Кандидат физико-математических наук, доцент
зав. кафедрой информационных технологий
Владимирского филиала Российской академии народного хозяйства
и государственной службы при Президенте Российской Федерации
И. В. Сидорова

Кандидат физико-математических наук, доцент
зам. директора Педагогического института по учебно-методической работе
Владимирского государственного университета
имени Александра Григорьевича и Николая Григорьевича Столетовых
В. А. Игонин

Якубович, Д. А. Издательская система LaTeX [Электронный ресурс] : учеб. пособие / Д. А. Якубович, Е. С. Еропова ; Владим. гос. ун-т им. А. Г. и Н. Г. Столетовых. – Владимир : Изд-во ВлГУ, 2019. – 327 с. – ISBN 978-5-9984-0956-1. – Электрон. дан. (15,2 Мб). – 1 электрон. опт. диск (CD-ROM). – Систем. требования: Intel от 1,3 ГГц ; Windows XP/7/8/10 ; Adobe Reader ; дисковод CD-ROM. – Загл. с титул. экрана.

Содержит 23 темы по основам верстки документов в издательской системе LaTeX. Материал систематизирован и проиллюстрирован многочисленными примерами.

Предназначено для проведения лекционных и практических занятий со студентами педагогических вузов по дисциплинам «Информационные технологии в научной деятельности», «Информационные технологии в образовании». Может быть использовано для организации самостоятельной работы студентов, курсов повышения квалификации педагогических кадров и самообучения.

Рекомендовано для формирования профессиональных компетенций в соответствии с ФГОС ВО.

Ил. 205. Библиогр.: 52 назв.

ISBN 978-5-9984-0956-1

© ВлГУ, 2019

© Якубович Д. А., Еропова Е. С., 2019

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	8
Глава 1. ИЗДАТЕЛЬСКАЯ СИСТЕМА LATEX	9
1.1. Введение.....	9
1.1.1. Концепции визуальной и логической разметки документов.....	9
1.1.2. Визуальные редакторы.....	9
1.1.3. MS Office.....	12
1.1.4. Apache OpenOffice	15
1.1.5. Adobe InDesign	17
1.1.6. Веб-ресурсы для разработки документов	18
1.1.7. WYSIWYG-конструкторы сайтов.....	19
1.2. Языки разметки	20
1.2.1. Виды разметок.....	21
1.2.2. Краткая история	21
1.2.3. Наиболее известные языки разметки.....	22
<i>Вопросы для самопроверки</i>	<i>28</i>
<i>Практикум</i>	<i>29</i>
1.3. TeX и LaTeX	30
1.3.1. Предпосылки развития системы TeX и LaTeX.....	30
1.3.2. Характеристика возможностей LaTeX	33
<i>Вопросы для самопроверки</i>	<i>34</i>
<i>Практикум</i>	<i>34</i>
1.4. Принципы работы LaTeX	35
1.4.1. Пакеты.....	35
1.4.2. Редакторы кода.....	35
1.4.3. Средства просмотра.....	37
1.4.4. Общая схема работы компилятора	38
<i>Вопросы для самопроверки</i>	<i>39</i>
<i>Практикум</i>	<i>40</i>

1.5. Использование LaTeX.....	40
1.5.1. Производительность и качество верстки	40
1.5.2. LaTeX в научной работе.....	44
1.5.3. LaTeX в практике учителя	49
<i>Вопросы для самопроверки</i>	52
<i>Практикум</i>	52
1.6. Практика использования LaTeX авторами.....	53
1.6.1. Аprobация материалов.....	53
1.6.2. Электронный учебник	56
1.7. Программное обеспечение	59
1.7.1. Основные инструменты разработчика	59
1.7.2. Дистрибутив	60
1.7.3. Редакторы кода.....	62
1.7.4. Средства просмотра.....	64
1.7.5. Компиляторы.....	65
1.7.6. Веб-сервисы для работы с LaTeX	66
<i>Вопросы для самопроверки</i>	68
<i>Практикум</i>	68
Глава 2. ОСНОВЫ LATEX.....	69
2.1. Структура документа	69
2.1.1. Hello, LaTeX!.....	69
2.1.2. Второй пример	70
<i>Вопросы для самопроверки</i>	72
<i>Практикум</i>	72
2.2. Работа с Overleaf.com.....	73
2.2.1. Регистрация аккаунта	73
2.2.2. Функции пользователя	75
<i>Вопросы для самопроверки</i>	81
<i>Практикум</i>	81
2.3. Обзор команд	83
2.3.1. Пример текста с математической символикой	83
2.3.2. Общее описание документа	86
2.3.3. Команды и символ \.....	86
2.3.4. Класс документа.....	86
2.3.5. Подключение пакетов	86
2.3.6. Титульный лист.....	87
2.3.7. Комментарии	88

2.3.8. Рубрикация документа	88
2.3.9. Текст и математические символы	89
2.3.10. Окружения	89
2.3.11. Перекрестные ссылки	90
<i>Вопросы для самопроверки</i>	91
<i>Практикум</i>	91
2.4. Настройка документа	94
2.4.1. Классификация команд LaTeX	94
2.4.2. Класс документа	100
2.4.3. Подключение дополнительных пакетов	102
2.4.4. Колонтитулы	104
2.4.5. Геометрия страницы	107
<i>Вопросы для самопроверки</i>	110
<i>Практикум</i>	110
2.5. Шрифт	112
2.5.1. Настройка начертания шрифта	112
2.5.2. Настройка цвета	117
<i>Вопросы для самопроверки</i>	120
<i>Практикум</i>	120
2.6. Абзацы	126
2.6.1. Абзац в структуре документа	126
2.6.2. Вертикальные и горизонтальные отступы	133
2.6.3. Заполнители	136
<i>Вопросы для самопроверки</i>	140
<i>Практикум</i>	140
Глава 3. МЕХАНИЗМЫ АВТОМАТИЗАЦИИ ОФОРМЛЕНИЯ	
И РАБОТА С ОБЪЕКТАМИ В LATEX	150
3.1. Рубрикация и оглавление	150
3.1.1. Команды рубрикации	150
3.1.2. Генерация оглавления	158
3.1.3. Оформление приложений	161
<i>Вопросы для самопроверки</i>	163
<i>Практикум</i>	163
3.2. Перечисления	171
3.2.1. Оформление перечислений	171
3.2.2. Пример оформления списков	174
3.2.3. Настройка отступов списка	175

<i>Вопросы для самопроверки</i>	177
<i>Практикум</i>	177
3.3. Изображения	179
3.3.1. Особенности работы с изображениями	179
3.3.2. Работа с изображениями	180
<i>Вопросы для самопроверки</i>	190
<i>Практикум</i>	190
3.4. Таблицы	195
3.4.1. Окружение tabbing	195
3.4.2. Окружение tabular	197
<i>Вопросы для самопроверки</i>	201
<i>Практикум</i>	201
3.5. Библиография и перекрестные ссылки	204
3.5.1. Библиография	204
3.5.2. Окружение thebibliography	204
3.5.3. Ссылки	207
<i>Вопросы для самопроверки</i>	213
<i>Практикум</i>	214
3.6. Математические тексты	219
3.6.1. Общие сведения о формулах	219
3.6.2. Оформление формул	222
3.6.3. Окружение equation	233
3.6.4. Разные символы	235
<i>Вопросы для самопроверки</i>	238
<i>Практикум</i>	238
3.7. Создание новых команд	246
3.7.1. Макрос \newcommand	246
3.7.2. Новые окружения	250
<i>Вопросы для самопроверки</i>	253
<i>Практикум</i>	253
3.8. Презентации	257
3.8.1. Стандартные средства разработки презентаций	257
3.8.2. Пакет Beamer	260
<i>Вопросы для самопроверки</i>	274
<i>Практикум</i>	274
3.9. Примеры шаблонов документов	283
3.9.1. Шаблон делового письма	283

3.9.2. Шаблоны научных статей	286
3.9.3. Шаблоны книг	294
3.9.4. Шаблоны презентаций	301
3.9.5. Другие шаблоны	301
<i>Практикум</i>	302
3.10 Рекомендации для самостоятельного изучения LaTeX.....	308
3.10.1. Печатные и электронные издания.....	309
3.10.2. Электронные ресурсы.....	310
3.10.3. Совместимость LaTeX с другими форматами документов.....	311
ЗАКЛЮЧЕНИЕ	312
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	313
ГЛОССАРИЙ	319
ОБ АВТОРАХ	326

ВВЕДЕНИЕ

LaTeX («латех») – свободно распространяемая настольная издательская система. Разработанный в 1977 году на базе TeX, пакет существенно расширен и широко применяется в современной типографии. LaTeX популярен в научных кругах и используется для верстки научных статей, монографий и презентаций, выдержанных в лаконичном академическом стиле. Однако эта издательская система прекрасно подойдет для верстки документов любого уровня сложности. LaTeX – достойная альтернатива существующим офисным пакетам, а по ряду направлений – даже существенно их превосходящая.

Учебное пособие посвящено анализу особенностей верстки в пакете LaTeX. В первой главе сравниваются две концепции разработки документов, рассматриваются возможности LaTeX, программное обеспечение для работы с системой.

Вторая глава раскрывает основы работы с текстом.

Третья глава посвящена механизмам автоматизации оформления, работе с объектами и математическими текстами.

Каждая тема содержит необходимый теоретический минимум, множество примеров и иллюстраций. Для формирования опыта работы с LaTeX теоретический материал сопровождается практическими заданиями.

Учебное пособие ориентировано на выработку навыков работы с системой LaTeX и прикладным программным обеспечением. Оно предназначено для учителей и преподавателей, стремящихся расширить навыки своей работы с профессиональными инструментами разработки документов, а также может быть рекомендовано для подготовки студентов педагогических вузов.

Глава 1

ИЗДАТЕЛЬСКАЯ СИСТЕМА LATEX

1.1 Введение

1.1.1 Концепции визуальной и логической разметки документов

Исторически сложилось два подхода к разработке бумажной и электронной документации. Первый – визуальное редактирование структуры и форматирования документа. Этот подход реализован через специальный визуальный интерфейс приложения, предоставляющего функции для работы с документом и отображающего его вид непосредственно во время работы. Современное прикладное ПО ориентируется именно на принцип визуальности и эргономичности процесса [6].

Второй, и первоначальный подход в деле верстки документа, предполагал использование языков разметки. Программы для работы с документацией на базе языка разметки используют специальные команды (теги), размечающие структуру документа и указывающие на форматирование его отдельных частей.

Несмотря на тенденцию к визуализации интерфейса приложений, оба подхода имеют свои достоинства и недостатки. В современном ПО языки разметки используются для описания структур и данных приложения. С другой стороны, приложения для работы с языками разметки получают более удобную эргономику и развитый функционал, двигаются в сторону визуального редактирования.

1.1.2 Визуальные редакторы

Визуальный редактор – это программное средство для реализации работы по визуальному оформлению страниц документа, шаб-

лонов и материалов по средствам технологии WYSIWYG (от англ. What You See Is What You Get, «что видишь, то и получишь») [5].

Концепция WYSIWYG является основой множества современных программ, начиная от редакторов HTML-страниц и заканчивая пакетами для разработки программного обеспечения, в которых подобным образом реализован процесс создания пользовательского интерфейса приложения. Очевидное преимущество WYSIWYG – непосредственное отображение результата уже на этапе разработки документа или интерфейса ПО.

Первой программой, поддерживающей WYSIWYG-концепцию, считают текстовый редактор Bravo, разработанный в компании Xerox PARC для компьютеров Alto в 1974 году (разработчики: Батлер Лемпсон (Butler Lampson), Чарльз Симони (Charles Simonyi) и др.). [6] Однако редактор не стал коммерческим продуктом (рис. 1.1).

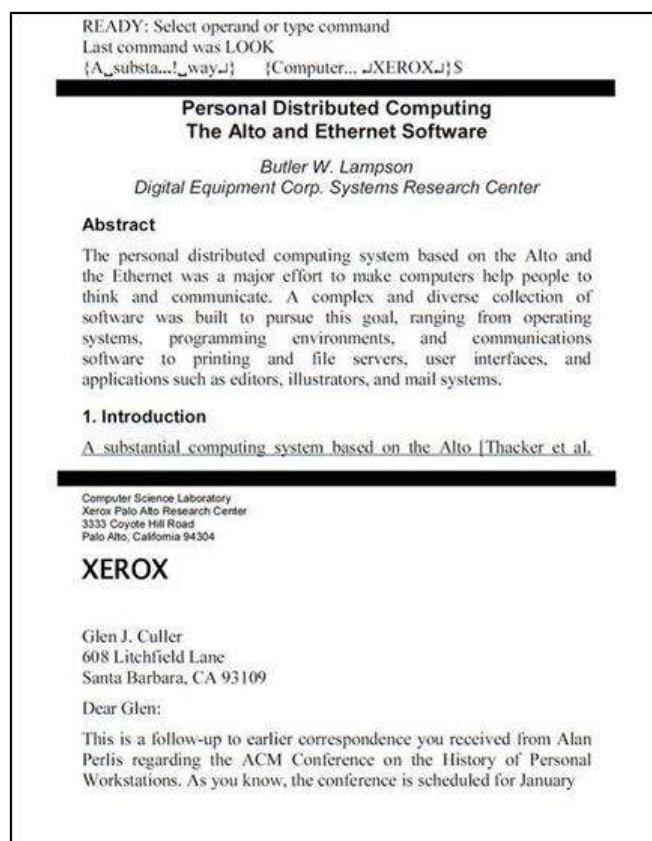


Рис. 1.1. Документ, набранный в редакторе Bravo.

Независимо от Xerox PARC, компания Hewlett Packard разработала и выпустила в конце 1978 года первую коммерческую программу, использующую принцип WYSIWYG – приложение для создания

диапозитивов (презентационной графики). Первые версии программы, названной BRUNO (рис. 1.2), работали на компьютерах HP 1000. Позже редактор был портирован на HP 3000 и переиздан как «HP Draw».



Рис. 1.2. Документ, набранный в редакторе BRUNO.

В 1970-х – начале 1980-х персональные компьютеры были редкостью, а также обладали весьма скромными графическими возможностями. Поэтому WYSIWIG-редакторы в основном использовались лишь на мощных рабочих станциях, доступных крупным коммерческим кампаниям.

К середине 1980-х ситуация изменилась: усовершенствование технологий позволило производить более доступные графические дисплеи, и WYSIWYG-программы начали появляться на более дешёвых и доступных широкому кругу покупателей компьютерах. Прежде всего – это популярный в свое время редактор LisaWrite (рис. 1.3) для компьютеров семейства Apple Lisa (1983 год) и MacWrite для Apple Macintosh (1984 год) [5].

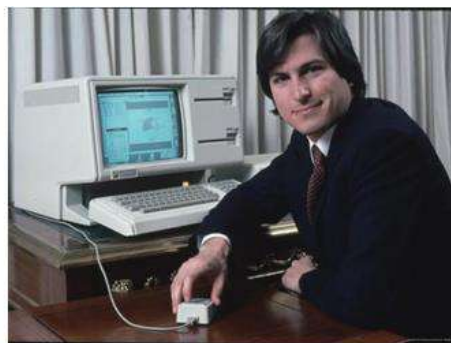
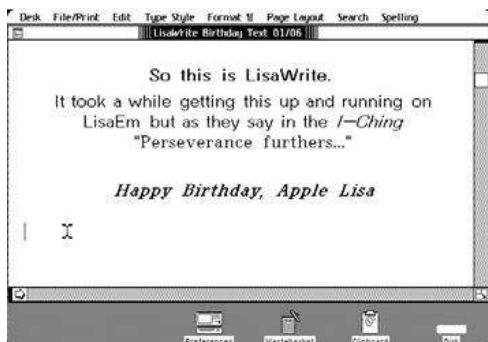


Рис. 1.3. Редактор LisaWrite, компьютеры Apple и основатель компании Apple Стив Джобс.

Развитие операционных систем семейства Windows в 1990-х годах привело к появлению пакета Microsoft Office [15]. Благодаря удобному визуальному интерфейсу и достаточно простой эргономике, пакет пользуется высоким спросом.

1.1.3 MS Office

Microsoft Office (MS Office) – офисный пакет приложений, созданных корпорацией Microsoft для операционных систем Microsoft Windows, Windows Phone, Android, OS X, iOS. В состав этого пакета входит программное обеспечение для работы с различными типами документов: текстами, электронными таблицами, базами данных и др.

Microsoft Office предоставляет целый ряд приложений с визуальным интерфейсом. Для разработки документов, презентаций, брошюр, листовок, публикаций (и других документов) используются пакеты MS Word, MS PowerPoint, MS Publisher и частично MS Excel. [15]

MS Word

Microsoft Word – текстовый процессор, предназначенный для создания, просмотра и редактирования текстовых документов, с локальным применением простейших форм таблично-матричных алго-

ритмов. Выпускается корпорацией Microsoft в составе пакета Microsoft Office [15,40].

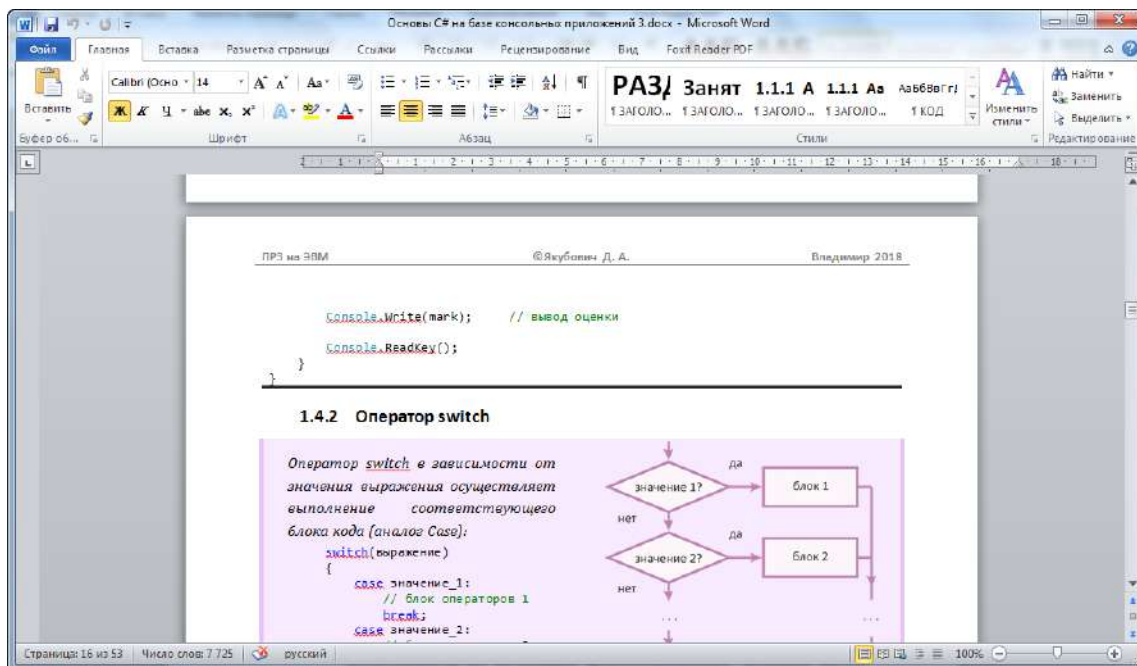


Рис. 1.4. Визуальный интерфейс MS Word 2010.

Достоинства:

- простой, понятный интерфейс;
- редактор WYSIWYG;
- огромная распространенность.

Недостатки:

- простой и интуитивно понятный интерфейс «провоцирует» пользователя на излишнее форматирование, которое часто приводит к ущербу качеству готового документа;
- проблемы и ограничения переносимости; особенно проблематично ситуация может обстоит в текстах технического или естественно-научного характера, содержащих большое количество формул;
- автоматическая нумерация рубрик, формул, изображений, библиографии, таблиц не предусматривается (ее можно организовать, однако пользователь должен знать механизмы работы стилей редактора);
- является проприетарным продуктом.

MS PowerPoint

MS PowerPoint – программа подготовки и просмотра презентаций, являющаяся частью Microsoft Office [15,41].

Microsoft PowerPoint поддерживает средства редактирования текста, форматирования и стилизации объектов, механизмы изменения расположения слайдов, их стилистической структуры.

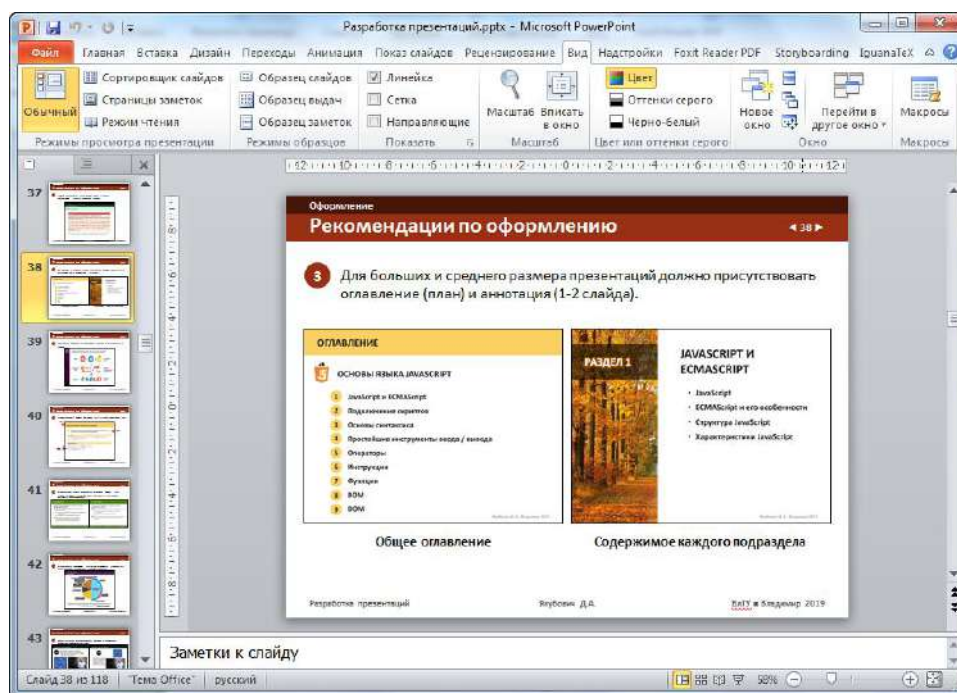


Рис. 1.5. Визуальный интерфейс MS PowerPoint 2010.

Достоинства:

- простой и эргономичный интерфейс пользователя;
- большая функциональность;
- обратная совместимость с разработчиками Office.

Недостатки:

- не предоставляет средств автоматической рубрикации;
- многие стандартные шаблоны оформления не учитывают психологические особенности восприятия информации;
- полностью поддерживается только на компьютерах с установленным пакетом MS Office.

MS Publisher

Microsoft Publisher – настольная издательская система, разработанная корпорацией Microsoft. В отличие от Microsoft Word, при-

ложение удобно для проектирования разметки страницы («черновика») [15,42].

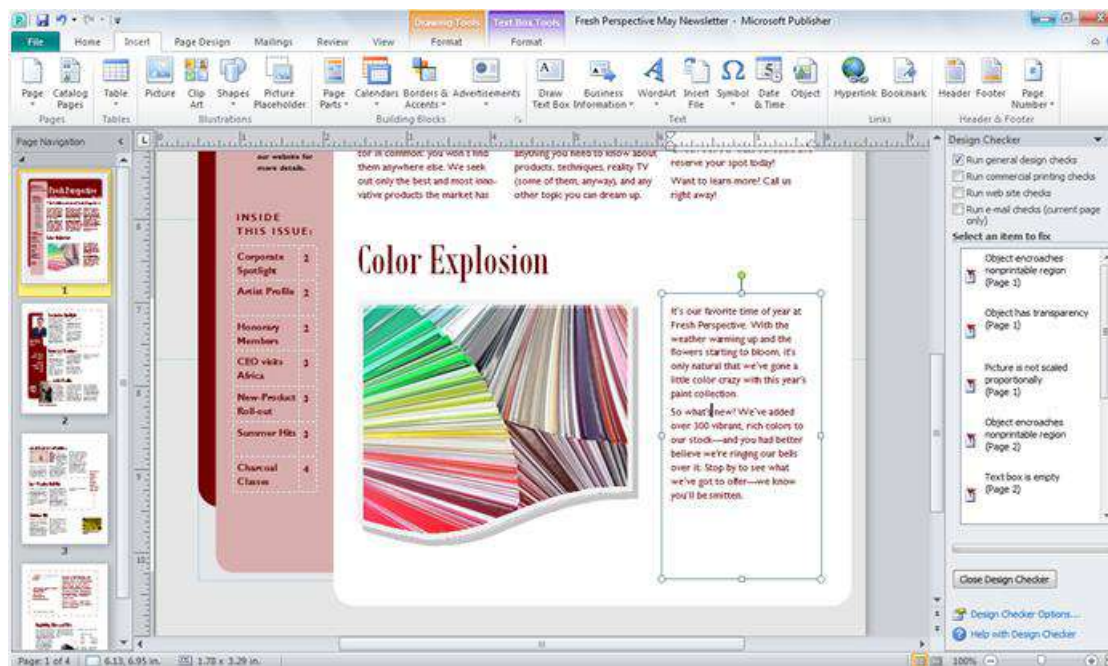


Рис. 1.6. Интерфейс Microsoft Publisher.

1.1.4 Apache OpenOffice

Apache OpenOffice (ранее OpenOffice.org, OO.org) – свободный пакет офисных приложений. Среди средств для визуального редактирования документов рассмотрим OpenOffice Writer и Impress [43].

В настоящее время Apache OpenOffice является ведущим пакетом офисного программного обеспечения с открытым исходным кодом, позволяющим разрабатывать текстовые документы, электронные таблицы, презентации, графические файлы, базы данных и др. Интерфейс приложения переведен на многие языки, работа с ним поддерживается на основных операционных системах. Также предоставляет возможность обработки данных офисных приложений других форматов. Возможности OpenOffice во много схожи с пакетом MS Office, однако имеются и существенные отличия.

Apache OpenOffice.org Writer

Apache OpenOffice.org Writer – текстовый процессор и визуальный (WYSIWYG) редактор HTML, входит в состав OpenOffice.org и является свободным программным обеспечением.

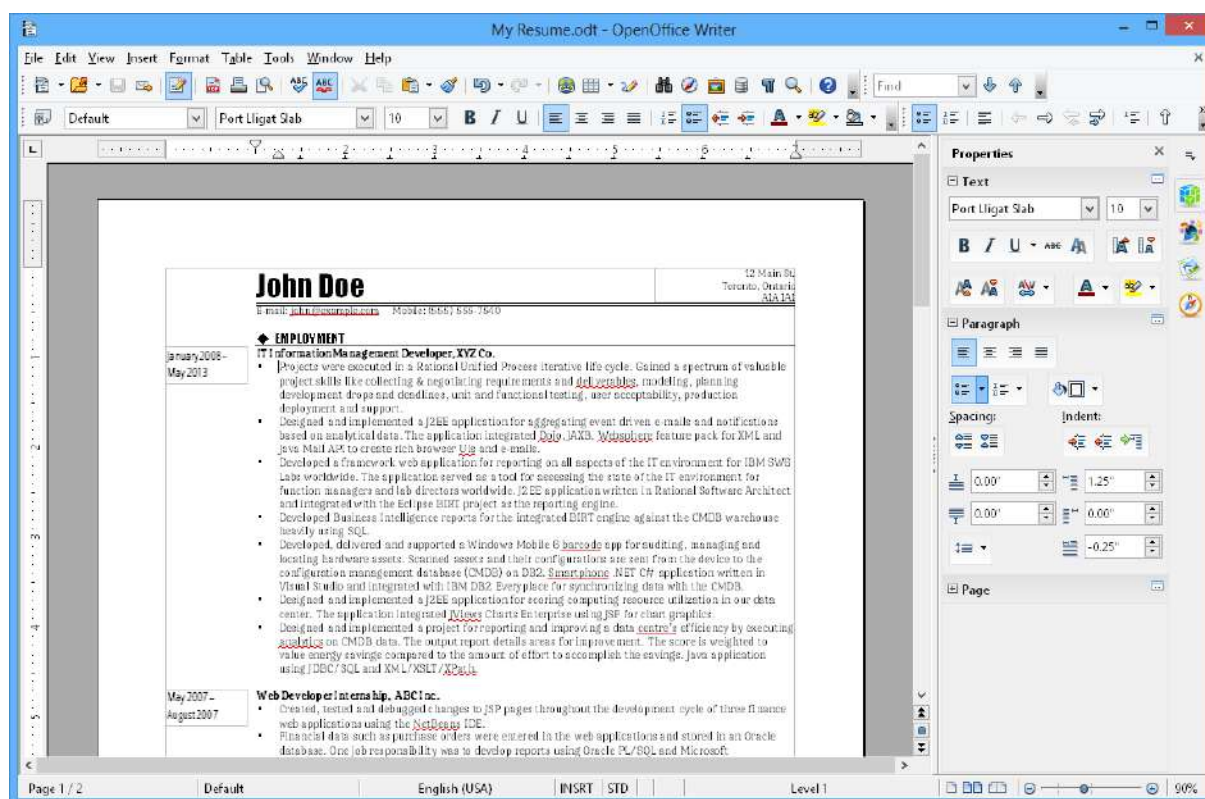


Рис. 1.7. Интерфейс редактора OpenOffice.org Writer.

Достоинства:

- является свободным аналогом коммерческого MS Word;
- обладает некоторыми возможностями, отсутствующими в MS Word, например, поддерживает стили страниц;
- возможность сохранять документы в различных форматах (DOC, RTF, XHTML, PDF и OASIS Open Document Format, который является форматом, используемым по умолчанию).

Недостатки:

- при работе со сложными электронными таблицами MS Excel в OpenOffice.org могут возникнуть проблемы;
- портируемость в более новый формат MS Word DOCX не поддерживается.

Apache OpenOffice Impress

Apache OpenOffice Impress – программа подготовки презентаций, входящая в состав OpenOffice. Способна экспортировать файлы презентации в форматы PDF SWF (Adobe Flash), что даёт возможность просматривать их на любом ПК с установленным Flash-проигрывателем. Может показывать, редактировать и сохранять фай-

лы в нескольких форматах, включая формат PPT, используемый в Microsoft PowerPoint.

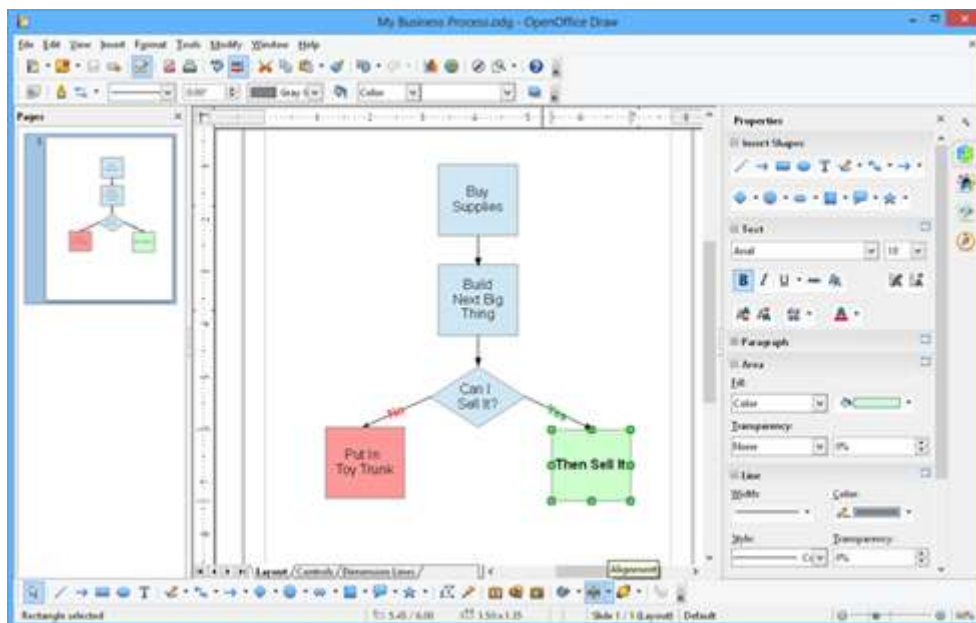


Рис. 1.8. Интерфейс редактора Apache OpenOffice Impress.

1.1.5 Adobe InDesign

Adobe InDesign – программа компьютерной вёрстки, разработанная фирмой Adobe Systems.

Adobe InDesign позволяет создавать высококачественные документы, поддерживает экспорт в разные форматы электронных изданий, в частности – в PDF. Установка официальной версии программы проходит автоматически. Является проприетарным продуктом [44].

Достоинства:

- развитый визуальный интерфейс;
- большой набор инструментов для профессиональной верстки документа.

Недостатки:

- проблемы совместимости версий;
- повышенный «порог входа» в систему;
- более высокое потребление ресурса ПК, чем у популярных настольных приложений.

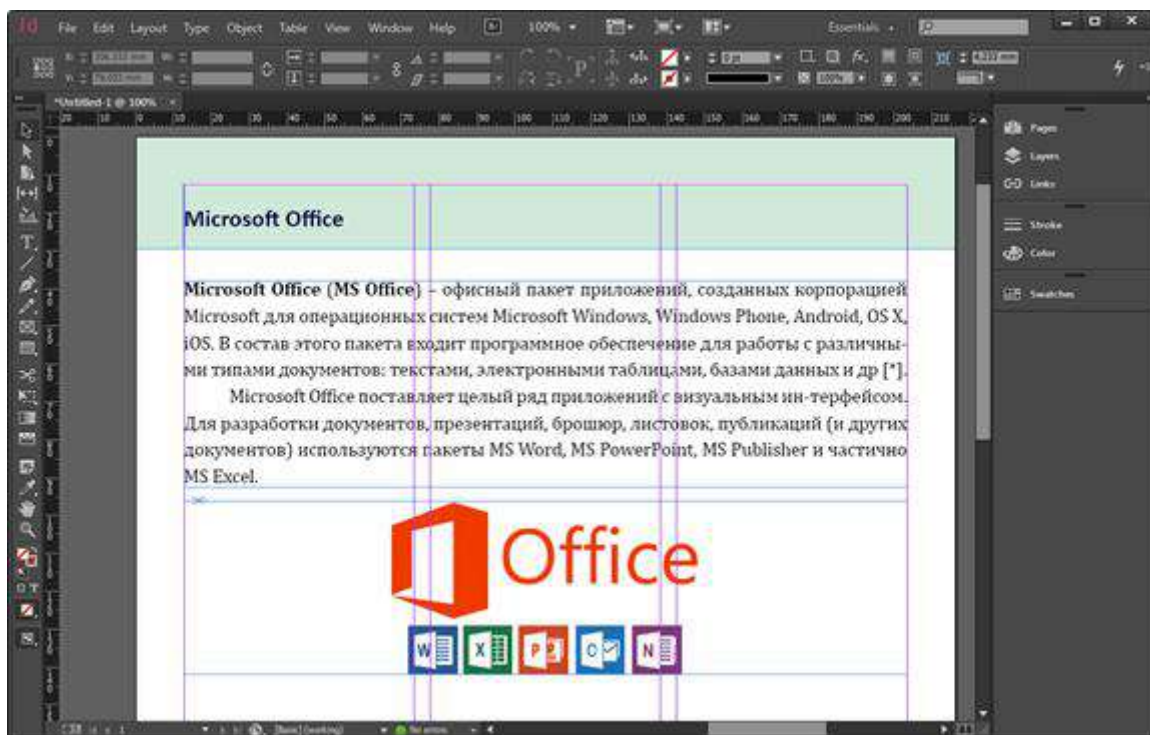


Рис. 1.9. Визуальный интерфейс системы компьютерной верстки InDesign CC 9.

1.1.6 Веб-ресурсы для разработки документов

Развитие сети Интернет приводит к появлению веб-сервисов для работы с электронными документами. Многие крупные компании (например, Microsoft), разрабатывающие ПО, постепенно начинают предлагать услуги удаленной работы с программами.

Достоинства удаленной работы:

- для работы достаточно наличия доступа к сети Интернет;
- не требуется установки специального ПО (работа через окно браузера);
- относительно скромные ресурсы ПК (т.к. вычислительные операции выполняются на стороне сервера).

Недостатки:

- может требоваться платная подписка на использование;
- часто возможности веб-ресурса не столь широки, как в аналогичной программе для ПК.

Среди известных веб-сервисов, позволяющих бесплатно создавать документы, презентации и электронные таблицы, отметим зарубежный Google и отечественный Яндекс.

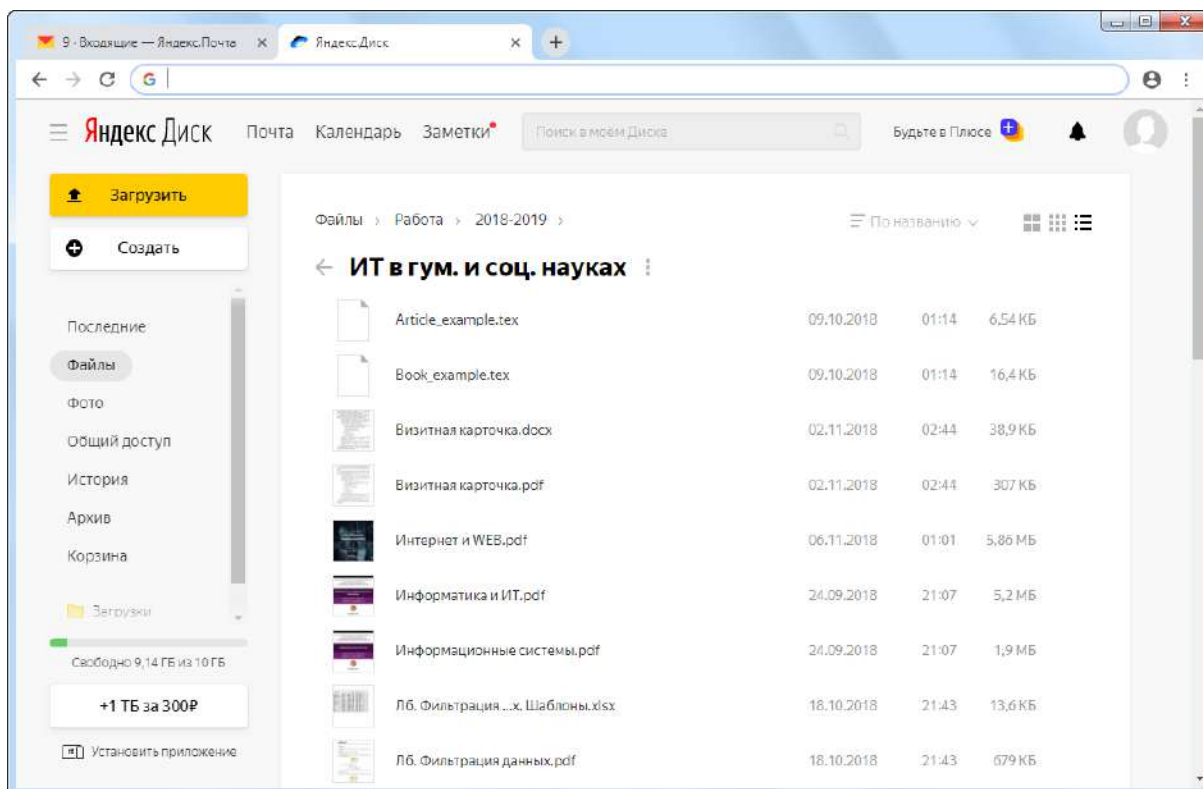


Рис. 1.10. Размещение документов на сервисе Яндекс Диск.

1.1.7 WYSIWYG-конструкторы сайтов

Современный веб-документ является существенным развитием классического понятия бумажного документа, включающим гипертекстовую технологию и мультимедиа.

Конструктор сайтов – прикладное программное обеспечение, реализующее инструменты по разработке веб-страниц в WYSIWYG-конструкторе [14].

Конструктор сайтов позволяет заранее определить шаблон страницы сайта (интернет-магазин, блог, электронный учебник, справочная система, визитка, другое), цветовое оформление, отображаемые модули. Использование конструктора сайта позволяет избежать изучения аспектов профессиональной работы с интернет-технологиями, веб-дизайном и веб-программированием.

Достоинства:

- простота использования и доступность непрофессиональному пользователю;
- разграничение содержимого и оформления;

- наличие готовых шаблонов под определенные типы проектов.

В качестве основного недостатка выступает ограничение в выборе стиля: он может быть современным, но типовым и не отражать в полной мере задумку автора.

Наиболее популярными конструкторами сайтов являются Wix.com (рис. 1.11), Drupal, Joomla!, Wordpress.

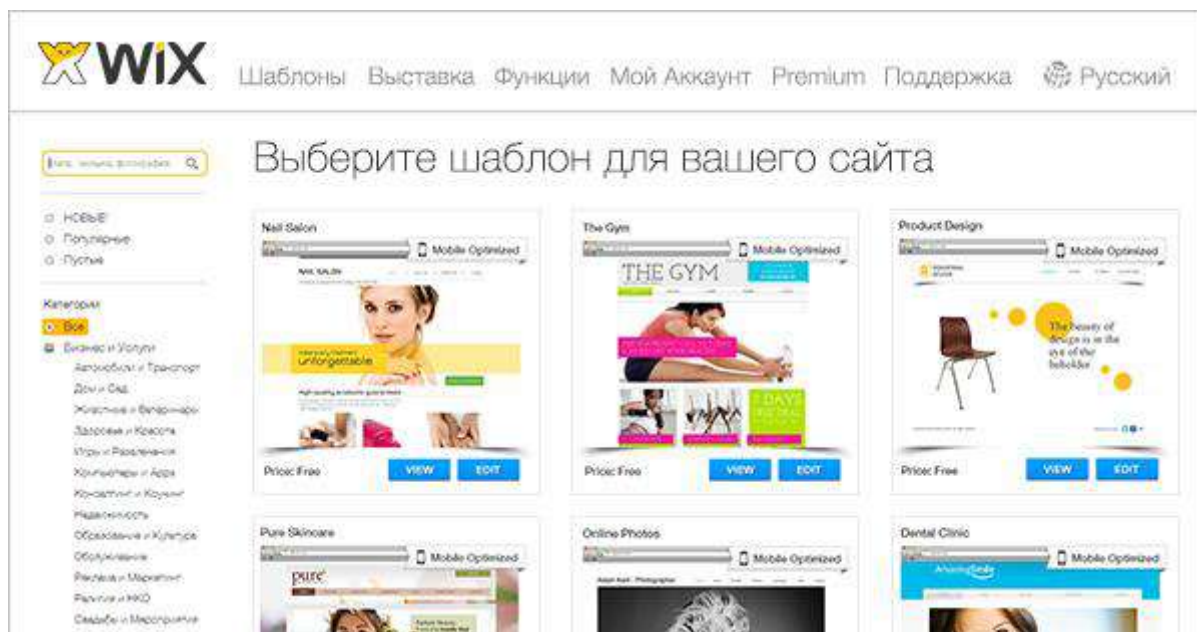


Рис. 1.11. Список шаблонов конструктора сайтов Wix.com.

1.2 Языки разметки

***Язык разметки (текста)** – это набор символов или последовательностей, содержащийся в тексте и передающий информацию о его структуре и форматировании [6].*

Основная задача специальных символов и команд – определить логическую структуру документа: заголовки, абзацы, термины, перечисления, цитаты, изображения и другие элементы документа. Более развитые языки разметки позволяют вставлять в документ интерактивные элементы (например, гиперссылки) и содержание других документов [14].

Различают

- логическую разметку;
- визуальную разметку.

1.2.1 Виды разметок

Логическая разметка определяет роль каждого участка текстовой разметки в общей структуре документа (например, заголовков, текстовый абзац, таблица, изображение, другое). **Физическая разметка** определяет форму отображения (форматирование) элемента документа (например, оформление шрифта: полужирный, подчеркнутый, красным цветом, с отступами слева и справа и т.д.).

Главная концепция языков разметки состоит в том, что визуальное отображение документа должно автоматически генерироваться из логической разметки и не зависеть от его непосредственного содержания. Это упрощает автоматическую обработку документа. В WYSIWYG-редакторах последнее правило часто нарушается: например, создавая документ в редакторе MS Word, пользователь может выделять цитаты курсивом, но нигде не указывать, что фрагмент является цитатой.

Текст, размеченный с помощью языка разметки, может выполнять следующие функции:

1. описывать данные для приложения;
2. описывать структуру документа и требования к его оформлению.

1.2.2 Краткая история

Впервые идея использовать языка разметки в обработке текстов была предложена Вильямом Танниклиффом в 1967 году. В 1970-е годы Танниклифф руководил разработкой стандарта GenCode. Тем не менее, родоначальником языков разметки часто называют исследователя IBM Чарльза Голдфарба. Основную концепцию Голдфарб изло-

жил в 1969 году и в том же году он принял участие в создании языка IBM GML. В 1986 году был разработан стандарт SGML, принятый организацией ISO [39].

Развитие операционных систем и повышение доступности программного обеспечения с визуальным интерфейсом привело к активному развитию WYSIWYG-редакторов, которые стали вытеснять многие языки разметки. Однако для больших работ со сложной структурой и специфическим форматированием по-прежнему используется разметка. Кроме того, на базе языков разметки формируются метаданные для приложений.

Первым языком разметки с четким разделением структуры и оформления документа стал Scribe (Брайан Рейд, 1980 год). Отличительной концепцией было использование стилей, отделенных от кода разметки. Scribe повлиял на разработку языка GML (позже SGML), языков HTML и LaTeX.

1.2.3 Наиболее известные языки разметки

Язык разметки SGML

Созданию GML в начале 80х способствовала идея, что разметка должна быть сфокусирована на структурных аспектах документа. Внешнее представление документа должно быть предоставлено интерпретатору. Язык SGML был разработан комитетом, возглавляемым Голдфарбом, и объединил ряд идей, включая уже существовавший GenCode.

SGML фиксировал синтаксис для включения разметки в текст, отдельно описывал разрешенные теги (технология DTD – Document Type Definition). Это позволяло разработчику создавать любую разметку. SGML можно считать метаязыком, который лег в основу множества других языков разметки.

Несмотря на широкую популярность языка разметки SGML и использование в крупных проектах, отмечалось ряд серьезных недостатков: язык громоздкий, трудный для изучения, излишне гибкий. Тем не менее этот язык разметки был положен в основу популярного и по сей день языка HTML.

Языки разметки HTML и XHTML

HTML (от англ. HyperText Markup Language – «язык гипертекстовой разметки») – стандартизированный язык разметки документов во Всемирной паутине [6,39].

В настоящее время подавляющее число веб-страниц описаны на языке HTML (или XHTML). Язык HTML интерпретируется браузерами и отображается на экранах мониторов, смартфонов или планшетов в надлежащей форме.

XHTML (от англ. eXtensible Hypertext Markup Language – расширяемый язык гипертекстовой разметки) – семейство языков разметки веб-страниц на основе XML, повторяющих и расширяющих возможности спецификации HTML4.

Сейчас развитие XHTML остановлено; рекомендуется использовать последнюю версию стандарта HTML – HTML5. До появления современной спецификации HTML5 язык определялся как приложение SGML.

Концепция HTML была построена Тимом Бернесом-Ли в 1991 на базе SGML. В HTML использовалась описательная разметка, что упрощало работу. Кроме того, это способствовало гибкости и расширяемости создаваемых документов. Ряд специалистов считают HTML контейнерным языком, поскольку правила расположения тегов требуют соблюдения иерархии их вложения.

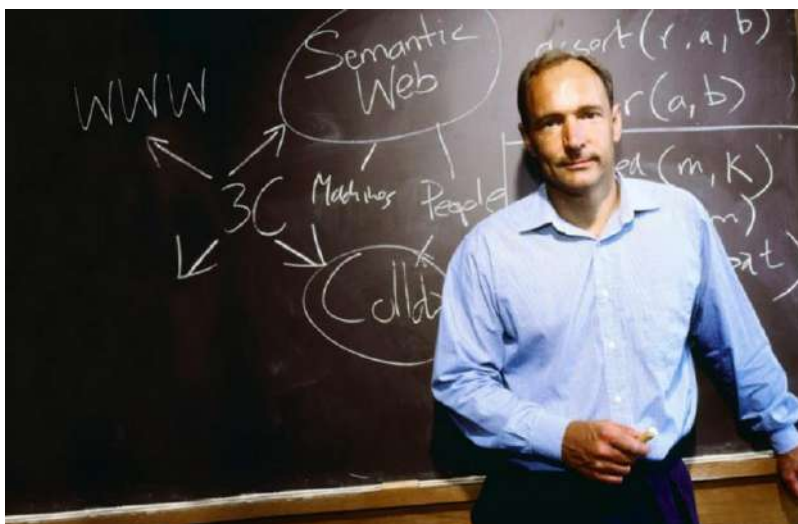


Рис. 1.12. Тим Бернес-Ли – разработчик HTML, технологий URI, URL, и HTTP, создатель Всемирной паутины (совместно с Робертом Кайо) и действующий глава W3C.

Начиная с января 2000 года была предложена аббревиатура XHTML. В отличие от HTML, спецификация XHTML четко фиксирует правило описания тегов и их атрибутов.

Язык разметки HTML используется для логического описания структуры веб-страниц. Основным элементом разметки является тег – это команда, которая определяет роль фрагмента документа.

Пример

Рассмотрим фрагмент HTML-разметки:

```
<body>
  <h1>Язык гипертекстовой разметки HTML</h1>
  <h2>Введение</h2>
  <p>
    <strong>HTML</strong> (от англ. HyperText Markup
    Language - язык гипертекстовой разметки) -
    стандартизированный язык разметки документов во
    Всемирной паутине. В настоящее время ...
  </p>
  
</body>
```

Очевидно, что текст разметки содержит специальные теги, обозначающие роль каждого текстового фрагмента:

Роль – заголовок первого уровня

Текст заголовка

```
<body>
  <h1>Язык гипертекстовой разметки HTML</h1>
  <h2>Введение</h2>
  <p>
    <strong>HTML</strong> (от англ. HyperText . . .
  </p>
  
</body>
```

Специальные команды логической разметки текста (**теги**)

Рис. 1.13. Теги в разметки веб-документа.

Теги не отображаются при выводе документа в браузере, однако браузер отображает текстовую и графическую информацию согласно назначению тегов:

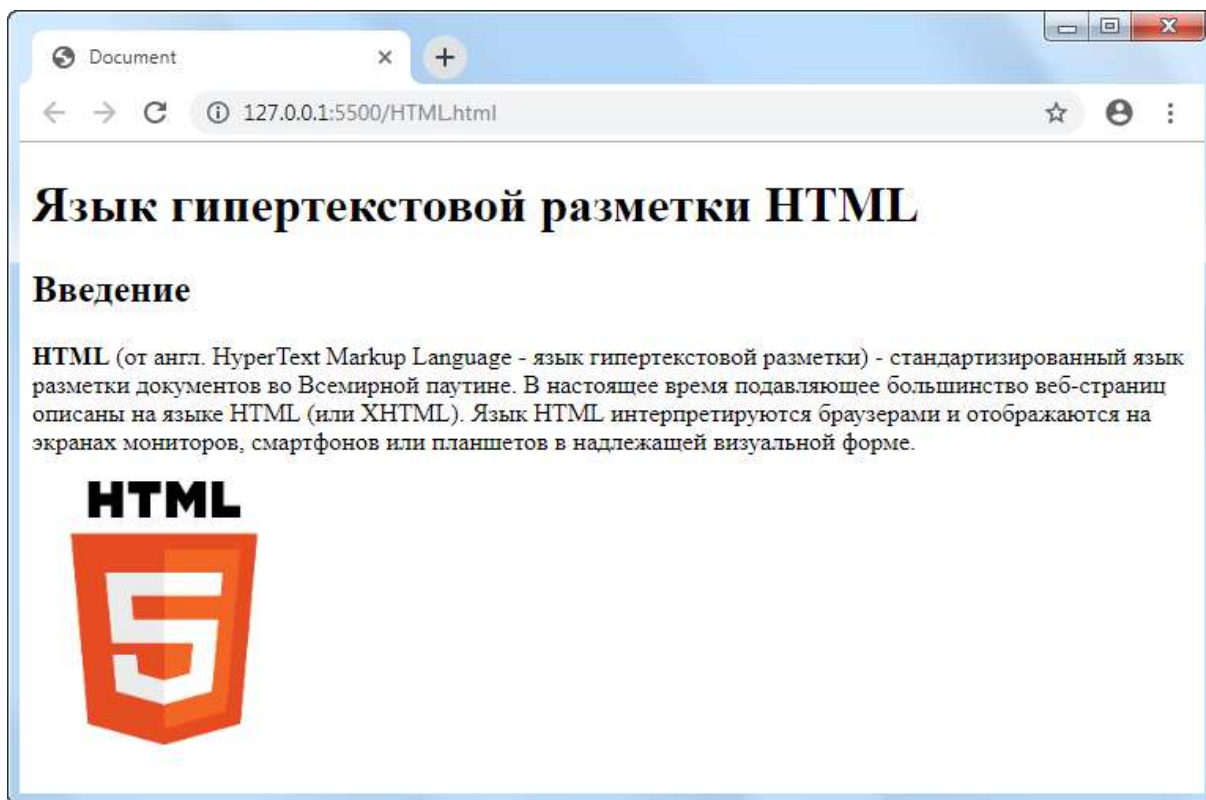


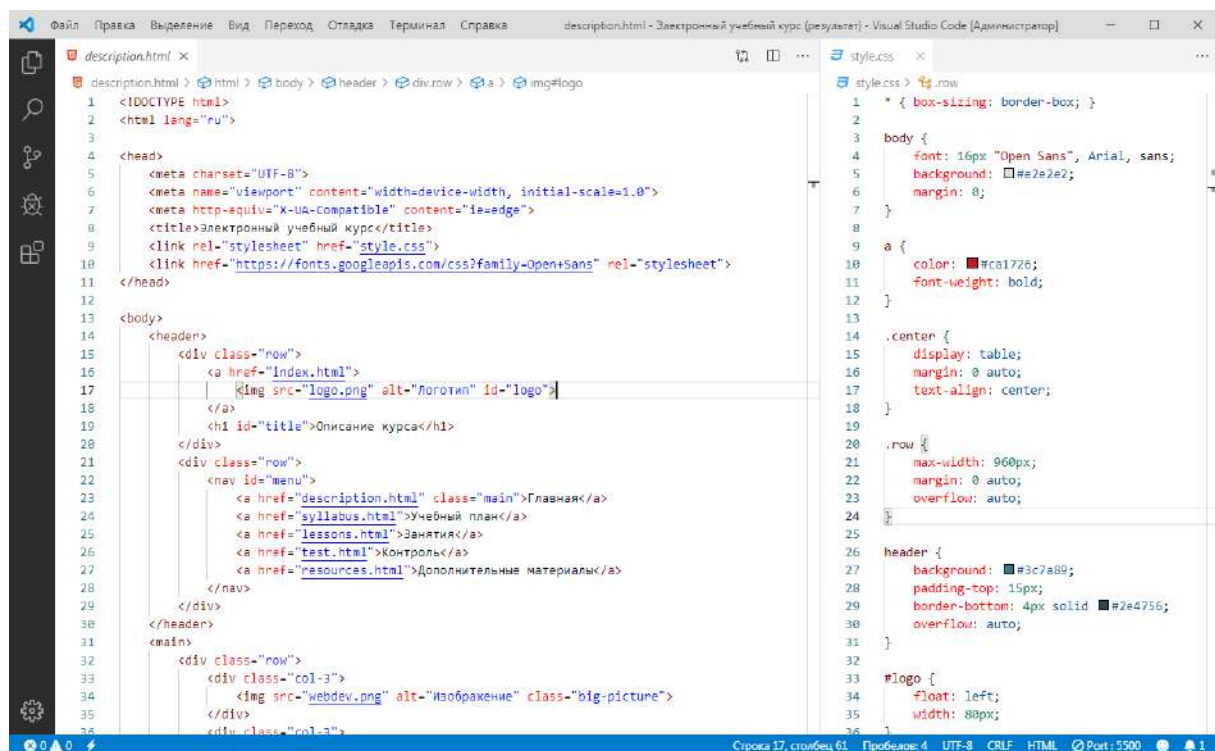
Рис. 1.14. Вывод HTML-документа в браузере.

Согласно рис. 1.14 за тегами закреплено определенное оформление текста. Любой из используемых пользователем браузеров будет отображать страницу аналогично.

Старые стандарты HTML содержали теги для физического форматирования текста, что нарушало саму концепцию HTML. Последняя спецификация HTML5 лишена этого недостатка.

Важно отметить, что оформление страницы с HTML-разметкой может быть совершенно любым! Для этого обычно совместно с HTML используется технология каскадных таблиц стилей (CSS). Она позволяет настроить оформление HTML-документа, не прибегая к изменению его логической структуры.

Кроме того, здесь демонстрируется важный принцип верстки документа – разделение логической и физической разметки. Это очень удобно, поскольку разработчик отдельно прорабатывает структуру документа и его дизайн. Для удобства современные текстовые редакторы поддерживают режим параллельного редактирования в отдельных колонках (окнах) кода HTML и CSS (см. рис. 1.15).



```

1 <!DOCTYPE html>
2 <html lang="ru">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <meta http-equiv="X-UA-Compatible" content="ie=edge">
8   <title>Электронный учебный курс</title>
9   <link rel="stylesheet" href="style.css">
10  <link href="https://fonts.googleapis.com/css?family=Open+Sans" rel="stylesheet">
11 </head>
12
13 <body>
14   <header>
15     <div class="row">
16       <a href="index.html">
17         
18       </a>
19       <h1 id="title">Описание курса</h1>
20     </div>
21     <div class="row">
22       <nav id="menu">
23         <a href="description.html" class="main">Главная</a>
24         <a href="syllabus.html">Учебный план</a>
25         <a href="lessons.html">Занятия</a>
26         <a href="test.html">Контроль</a>
27         <a href="resources.html">Дополнительные материалы</a>
28       </nav>
29     </div>
30   </header>
31   <main>
32     <div class="row">
33       <div class="col-3">
34         
35       </div>
36       <div class="col-3">
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

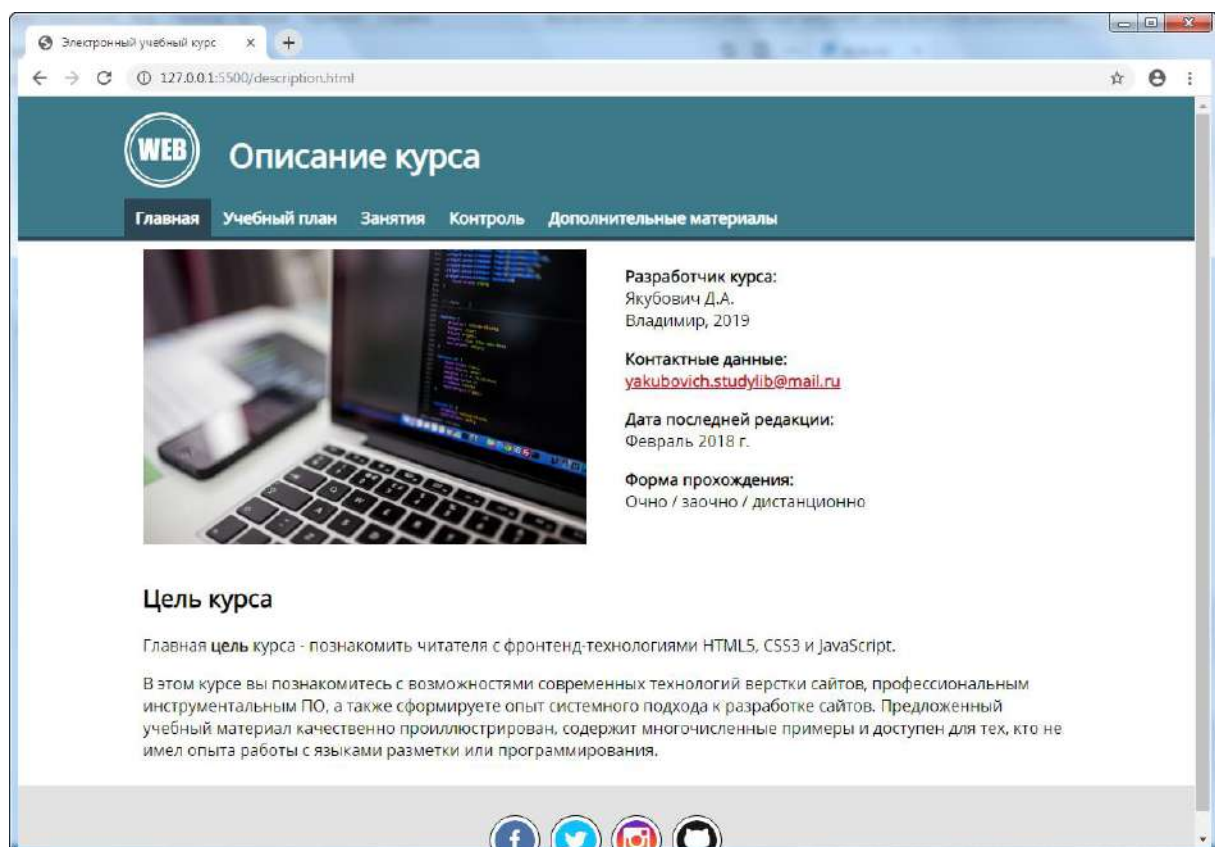


Рис. 1.15. Разделение разметки веб-страницы на HTML-документ и CSS-стили. Для подготовки использован текстовый редактор Visual Studio Code. Окончательный вид веб-страницы в браузере.

Язык разметки XML

XML (от англ. eXtensible Markup Language, расширяемый язык разметки) – метаязык, используемый для создания и обработки документов программами и одновременно удобный для чтения и создания документов человеком [6,39].

XML разработан консорциумом W3C. Изначально язык фокусировался на описание документов в сети Интернет.

В отличие от HTML, XML допускает создание любого тега. Однако XML лишился многих ориентированных на людей особенностей языка SGML, поэтому основное его назначение предполагалось в области компьютерной обработки автоматически генерируемых данных. В настоящее время XML широко используется как формат хранения данных и описания их структур. Язык имеет четкие правила описания тегов. Как и HTML, он может быть охарактеризован как «контейнерный» язык.

Пример

С помощью тегов XML опишем данные о студенте: укажем номер зачетной книжки, имя, фамилию, возраст, название группы и телефонный номер:

```
<student>
  <id>17085</id>
  <fullname>
    <firstname>Петр</firstname>
    <lastname>Краснов</lastname>
  </fullname>
  <age>20</age>
  <group>МИ-116</group>
  <phonenumber>1 234 5678900</phonenumber>
</student>
```

По вложению тегов можно установить структуру записи и отношение элементов. Далее такая запись может быть обработана программой и, например, занесена в базу данных. Разумеется, в одном документе может быть перечислено любое число таких записей, соответствующих описанию разных студентов.

Язык разметки JSON

JSON (от англ. JavaScript Object Notation) – текстовый формат обмена данными, основанный на языке программирования JavaScript.

Формат допускает простое чтение людьми. JSON разработан Дугласом Крокфордом [39].

JSON совместим с XML, но более лаконичен.

Формат JSON не зависит от языка JavaScript и может использоваться практически с любым языком программирования (в частности, ряд языков программирования содержат встроенные функции для работы с JSON).

JSON-разметка может быть задана двумя способами:

- набором пар ключ: значение;
- упорядоченным набором значений.

Пример

Как и ранее, опишем данные о студенте:

```
{
  "student": {
    "id": "17085",
    "fullname": {
      "firstname": "Петр",
      "lastname": "Краснов"
    },
    "age": "20",
    "group": "МИ-116",
    "phonenumber": "1 234 5678900"
  }
}
```

Обратите внимание, что в описании четко прослеживается структура записи в форме «ключ-значение», и понятно, что студент – это Краснов Петр, 20 лет, учится в группе МИ-116, контактный номер 1 234 5678900, уникальный номер (зачетной книжки) – 17085:

Важно отметить, что делать отступы и писать ключи (теги) на отдельных строках необязательно: программы могут обрабатывать HTML-, XML и JSON-записи даже в одну строку (делается это для сжатия размеров файлов). Такая форма важна для человека, поскольку хорошо читаема.

Вопросы для самопроверки

1. Какие основные функции вкладываются в понятие WYSIWYG-редактора?

2. Приведите примеры известных вам визуальных редакторов для
 - верстки документов;
 - верстки презентаций;
 - верстки веб-сайтов;
 - разработки визуального интерфейса приложения.
3. Перечислите основные отличия концепции визуального редактирования документа от создания его на базе языка (логической) разметки.

Практикум

1. Недостатки визуальных редакторов

1. Каким визуальным текстовым редактором вы пользуетесь при подготовке документов (MS Office, OpenOffice Writer, другой)?
2. Опишите в таблице проблемы и недостатки, с которыми вы чаще всего сталкиваетесь при работе с вашим редактором:

№	Недостаток или проблема
1	...
2	...
3	...

3. Охарактеризуйте выбранный вами редактор как специалист в своей учебной дисциплины (научной отрасли). Какие объекты документа, на ваш взгляд, редактор позволяет оформить не так качественно, как ожидается?

2. Языки разметки

1. Приходилось ли вам работать с языками разметки? Если да, то охарактеризуйте процесс в сравнении с работой в визуальном редакторе.
2. В сети Интернет найдите любой простой пример разметки документа на языке HTML. Можете ли вы интуитивно дать описание хотя-бы некоторым элементам разметки? (Предполагается, что вы ранее не сталкивались с подобным)
3. Как вы считаете, в каком случае логическая разметка документа может быть более информативна, чем документ в визуальном редакторе?

1.3 TeX и LaTeX

1.3.1 Предпосылки развития системы TeX и LaTeX

TeX («tex») – система компьютерной вёрстки, разработанная американским профессором информатики Дональдом Кнутом в целях создания электронной компьютерной типографии [3].

Создание Д. Кнутом новой издательской системы было обусловлено качеством печати технической документации: использовавшийся в то время метод монотипии давал оттиски плохого качества (1977 год). Примерно в это же время Кнут впервые увидел результат работы высококачественной цифровой типографической системы и заинтересовался возможностями цифровой типографии. 13 мая 1977 года Кнут формулирует базовые возможности TeXa. Финальная версия системы вышла в 1979 году [4].



Рис. 1.16. Кнут, Дональд Эрвин (род. 1938 г.).

TeX содержит инструменты рубрикации документа, работы с перекрёстными ссылками, автоматической генерации оглавления и библиографии [9,10]. Высокое качество оформления математических формул и математической символики в целом обеспечили широкое распространение TeX в академических кругах. Его можно использовать для создания любого документа, начиная от делового письма и заканчивая сборкой многотомной книги [7].

Крупные научные издательства используют TeX для книгопечатания или книжного набора.

В отличие от обыкновенных текстовых процессоров, WYSIWYG-редакторов и систем компьютерной вёрстки, TeX-документ представляет собой текстовый файл, содержащий текст документа и специальные команды, определяющие его структуру и форматирование. TeX самостоятельно формирует окончательный вид документа на основе шаблона (класса документа), который указывает пользователь.

Отличительной чертой TeX является свободное распространение (лицензия Project Public License).

Высокое качество документа достигается благодаря специальной технологии для работы с шрифтом – METAFONT, описанной в работе Д. Кнута. Сам Кнут отмечал, что во многом качественное оформление документа определяется, прежде всего – качественным шрифтом [11].

При всех достоинствах, TeX является полноценным языком программирования. Он требует профессиональных навыков работы верстальщика, большого труда и объема работы при создании документа. Поэтому TeX имеет высокий порог вхождения в систему. Обширной аудитории пользователей столь тонкие знания не требуются. Поэтому для TeX были разработаны специальные упрощенные макропакеты, где одна команда заменяет серию команд TeXa. Пользователю достаточно знать ее назначение и не нужно задумываться о том, как она работает [8].

Наиболее популярным набором макропакетов является система LaTeX.

LaTeX («латех») – наиболее популярный набор макрорасширений системы компьютерной вёрстки TeX, который облегчает набор сложных документов. Система разработана в начале 80х Лесли Лампортом [12].



Рис. 1.17. Логотип LaTeX.

Важно заметить, что ни один макропакет не расширяет функционал TeX: всё, что можно сделать в LaTeX, реализуется и с помощью TeX (обратное неверно).

Первая версия LaTeX была выпущена Лесли Лэмпортом в 1984 году. Высокая популярность и необходимость интернационализации привела к появлению множества различных форматов. Остро потребовалось их упорядочить. Промежуточная версия LaTeX2ε (1993 год) объединила их в один и сохранила совместимость со старыми версиями LaTeX. В настоящее время LaTeX2ε остается основной версией пакета LaTeX [12,13].

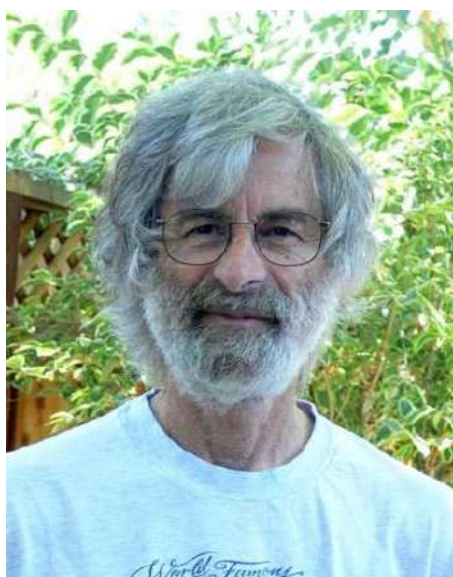


Рис. 1.18. Лэсли Лэмпорт – разработчик системы LaTeX.

В конце 80х TeX и LaTeX начали распространяться на территории СССР. Разработан автоматический перенос русских слов. Отдельной проблемой стало создание кириллических шрифтов: в силу меньшей популярности LaTeX в России приходится ограничиваться и меньшим числом доступных шрифтов.

1.3.2 Характеристика возможностей LaTeX

Преимущества

1. Четкая система структуризации разделов документа.
2. Автоматическая нумерация разделов, изображений, таблиц, расстановка ссылок на литературу и перекрестных ссылок, переносов.
3. Развитый механизм генерации библиографии, автоматическое создание оглавления одной командой.
4. LaTeX автоматизирует подготовку алфавитного указателя.
5. Высочайшее качество подготовки математических, физических, химических и биологических текстов с учетом принятых в полиграфии стандартов, а также любого документа научного характера (в т.ч. гуманитарного направления).
6. Автоматическая система «умной» компоновки и балансировки элементов документа в зависимости от пространства листа и заданного типа документа.
7. Позволяет определять на странице несколько министраниц, причем на каждой из них допустимы сноски.
8. Большой набор инструментов для вставки диаграмм, графов, схем, алгоритмов в векторной форме.
9. Широкий спектр возможных документов: от простых деловых писем до статей, докладов и книг высочайшего полиграфического качества.
10. Владение LaTeX является показателем профессиональной компетентности научного работника любой отрасли (тем более работника сферы образования).
11. Является свободно распространяемым ПО.

Недостатки

1. В процессе компиляции файла возможно потребление значительных машинных ресурсов, больших, чем у примитивных текстовых процессоров.
2. Высокое качество типографии требует и хорошего качества печатного оборудования.

3. LaTeX великолепен и прост для стандартных типов документов, однако разметка нестандартного документа требует профессиональных навыков.

Необходимо отметить, что перечисленные недостатки являются условными:

1. современные компьютеры и прикладное ПО существенно ускоряют процесс компиляции файлов;
2. современное печатное оборудование дает высокое качество печати;
3. нестандартные форматы документов крайне редко требуются в науке и образовании.

Вопросы для самопроверки

1. Что явилось причиной появления системы TeX?
2. Почему рядовому пользователю рекомендуется использование LaTeX вместо «чистого» TeX?
3. Перечислите основные возможности издательской системы LaTeX.

Практикум

Оформите краткие ответы на вопросы.

1. Что позволяла делать система TeX?
2. Изучите основные возможности системы LaTeX, используя материалы этого курса и открытые источники сети Интернет. Перечислите те достоинства системы, которые могут оказаться полезными вам в работе с документами.
3. Что из перечисленного не поддерживает ваш текущий редактор? Составьте таблицу сравнения используемого вами текстового редактора и LaTeX:

	Критерий	Используемый редактор	LaTeX
1	Система структуризации разделов	+	+
2	Автоматическая нумерация рубрик	настраиваемая	+
3

1.4 Принципы работы LaTeX

Главная концепция LaTeX – автор должен сконцентрироваться на содержании документа, а издательская система – на форматировании. Автор набирает логическую структуру текста, разбивая его специальными командами на главы, разделы, абзацы, таблицы, изображения, а LaTeX самостоятельно строит документ по стандартам оформления выбранного стиля. Это позволяет отделить содержание от оформления [31].

Документ, набранный в LaTeX, определяется стилевым файлом (шаблоном оформления). Существует несколько стандартных стилей для оформления научных статей, рефератов, докладов, книг, деловых писем, презентаций и других типов документов. Кроме стандартного набора пакетов, многие издательства, журналы, научные и учебные организации поставляют собственные стилевые файлы, что позволяет быстро оформить публикацию согласно стандартам издания [25,32].

1.4.1 Пакеты

LaTeX – это набор специальных **пакетов (библиотек классов)**, в которых реализован функционал для верстальщика. Для работы с LaTeX необходима установка специального дистрибутива, например, TeX Live, MikTeX, MakTeX (подробнее рассматриваются на занятии 1.4), либо использование веб-сервиса ([Overleaf.com](https://overleaf.com), [Sharelatex.com](https://sharelatex.com), [Latexbase.com](https://latexbase.com)).

1.4.2 Редакторы кода

Основным форматом LaTeX-файлов является **.tex**. Этот формат можно открыть и изменить содержимое в любом текстовом редакторе, например блокноте. Для работы с документами обычно используют продвинутые текстовые редакторы разработчика, например, Notepad++, Vim, Emacs, Sublime Text, Visual Studio Code. Они поддерживают подсветку синтаксиса языка, команды структуризации, возможность компиляции файлов и расширенные возможности текстовых редакторов [14].

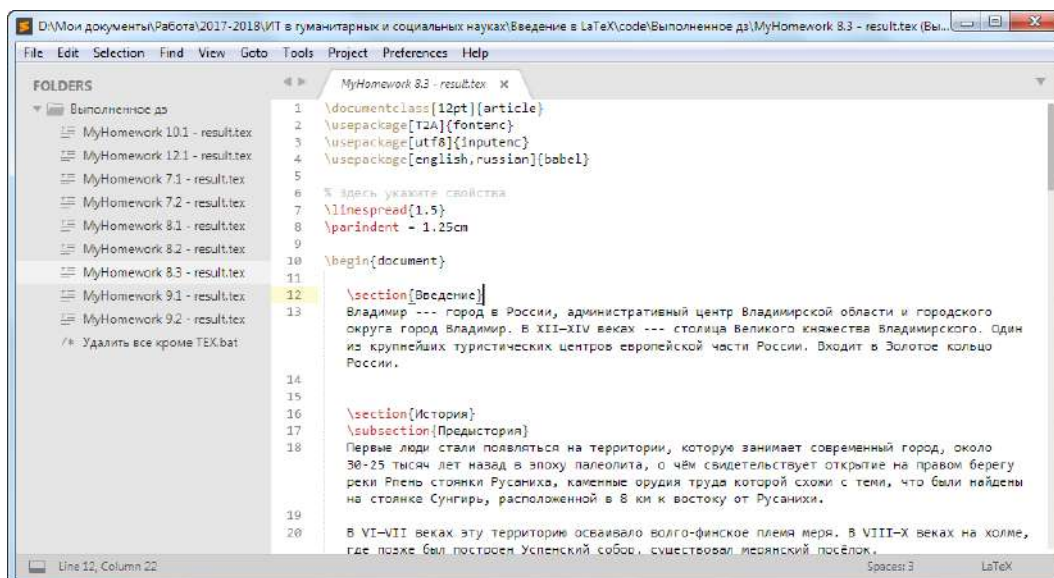


Рис. 1.19. Редактирование LaTeX-документа в редакторе Sublime Text 3.

Кроме текстовых редакторов, используются среды разработки, совмещающие одновременно многофункциональный текстовый редактор, элементы визуального форматирования и автоматические средства сборки документа. Среди наиболее развитых систем отмечают TeXworks, TeXstudio и WinEdt (см. занятие 1.7).

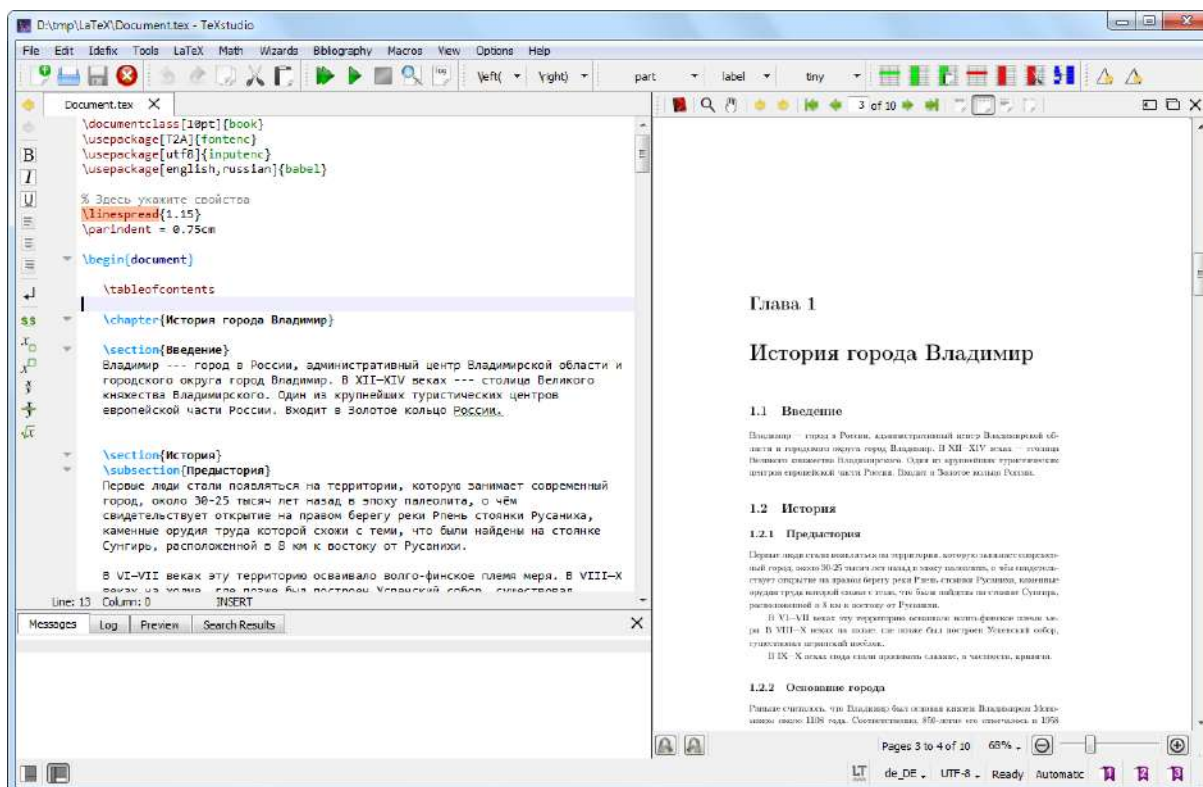


Рис. 1.20. Фрагмент LaTeX-документа, открытого в среде разработки TeXstudio.

1.4.3 Средства просмотра

С помощью программы компилятора `tex`-файл транслируется в другой формат, который допускает визуальное отображение документа. Обычно компилятор изначально «вшит» в среду разработки или дистрибутив пакетов `LaTeX`, дополнительной настройки не требуется [5].

В качестве одного из промежуточных типов файлов, допускающих просмотр конечного вида документа, используют формат `DVI` — это не зависящий от устройства (от англ. *DeVice Independent*) формат, который хранит информацию о форматировании текста и размещении всех его элементов на странице, но без самих букв и изображений. Программы, преобразующие `dvi`-файл в другой формат, называются `dvi`-драйверами, например:

- **xdvi** — преобразует `DVI`-файл в изображение;
- **dvips** — преобразует в `PostScript`, который можно передать принтеру на печать;
- **dvipdf** — транслирует `DVI` в `PDF`.

Одной из программ подобного рода является `YAP` (см. рис. 1.21).

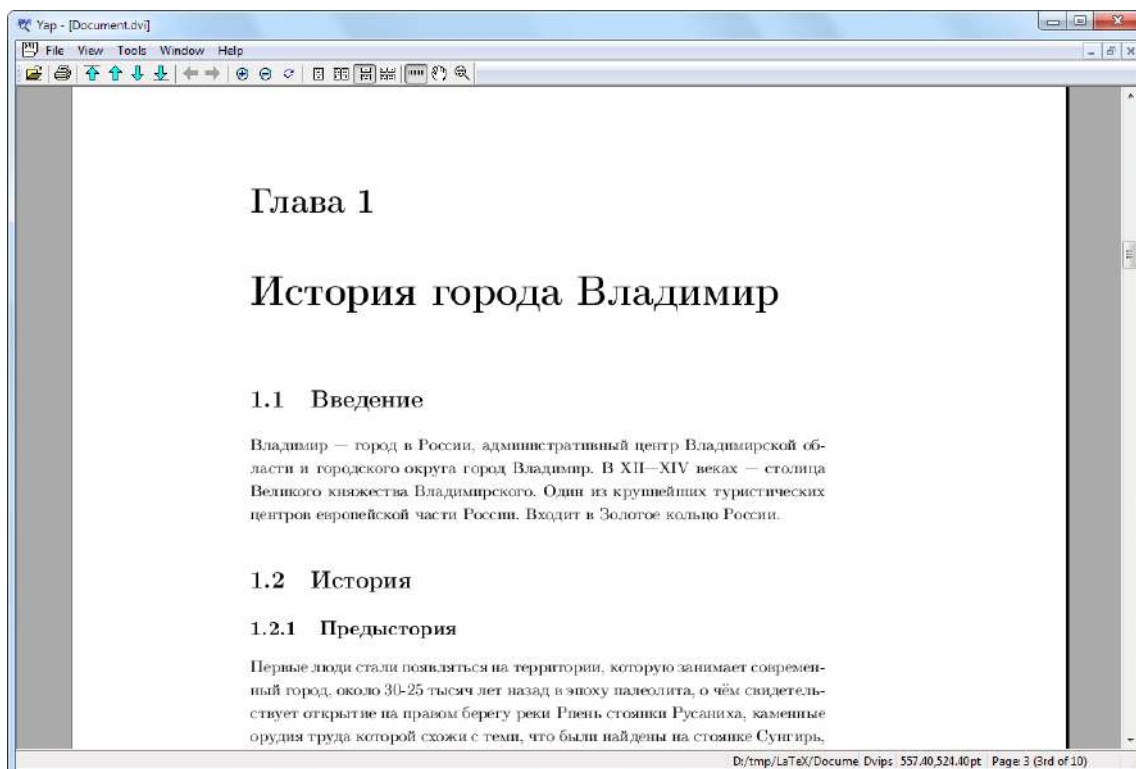


Рис. 1.21. `YAP` — одна из программ для просмотра `DVI`-файлов.

1.4.4 Общая схема работы компилятора

Кроме указанных типов файлов, в процессе работы компилятора LaTeX могут генерироваться промежуточные файлы, являющиеся конечным результатом или передающие компилятору данные для повторного прохода, например:

- .log – файл протокола; в этот файл записывается вся информация, имеющая отношение к компиляции.
- .aux – информация о перекрёстных ссылках;
- .toc – файл оглавления (table of contents);
- .lof – список иллюстраций (list of figures);
- .lot – список таблиц (list of tables);
- .bbl – список литературы, сформированный с помощью программы BibTEX;
- .ind – предметный указатель, сформированный программой MakeIndex.

Поскольку компиляция LaTeX документа концептуально аналогична компиляции программного кода, то промежуточные файлы позволяют ускорять процесс обработки документа за счет повторного использования данных. После завершения работы над проектом промежуточные файлы можно удалить [11,13].

Процесс работы компилятора можно представить следующей схемой (подробнее изложено в работах [5,17]):

1. Компилятору передается исходный файл (или несколько файлов) с кодом разметки.
2. Дополнительно могут запрашиваться внешние пакеты, классы, описание шрифтов.
3. При необходимости осуществляется повторный проход компилятора (например, для индексации перекрёстных ссылок, таблиц, изображений и т.д.).
4. Создается файл формата DVI. Также может быть создан файл DVIPS или PS, в зависимости от того, какой формат конечного документа будет выбран пользователем для дальнейшего просмотра.
5. DVI преобразуется в формат PDF (см. рис. 1.22) как наиболее универсальный формат для просмотра документа.

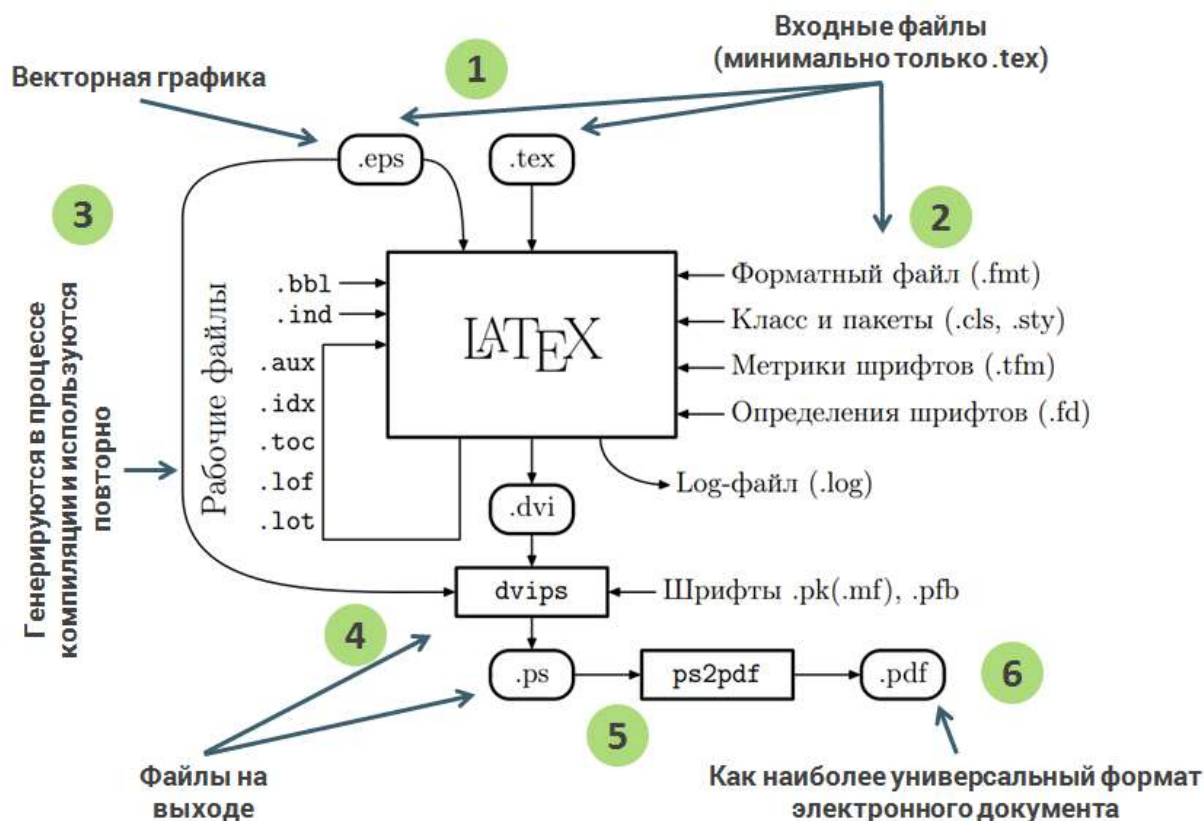


Рис. 1.22. Схема работы LaTeX (номераи помечены этапы работы).

В качестве наиболее универсального формата отображения конечного результата используется формат PDF.

Вопросы для самопроверки

1. Что представляет собой основной документ в формате TeX? Опишите общую процедуру создания документа в этом формате.
2. Что понимают под пакетом или классом документа?
3. В ких редакторах можно работать с LaTeX? Можно ли для этого использовать стандартный Блокнот?
4. Зачем нужны программы просмотра документа?
5. Опишите общий механизм работы компилятора LaTeX.
6. Зачем LaTeX дополнительные файлы при компиляции документа?

Практикум

1. В сети Интернет найдите описание текстового редактора Notepad++. Скачайте его с официального сайта и установите на свой ПК.
2. Сравните возможности Notepad++ и обычного приложения Блокнот.
3. Возьмите любой файл с расширением .tex и откройте его в Notepad++. По каким признакам можно сказать, что это текст разметки?
4. Найдите в сети интернет описание редактора Visual Studio Code. Попытайтесь сравнить его с Notepad++.
5. Попробуйте открыть .tex-файл в MS Word.

1.5 Использование LaTeX

1.5.1 Производительность и качество верстки

Выбор программного обеспечения для подготовки учебной или научной документации может определяться рядом факторов:

- функциональностью пакета офисных приложений;
- качеством конечного продукта;
- уровнем сложности и доступности для пользователя («порогом вхождения»), в том числе установкой исходных файлов и конфигурировании системы;
- лицензией (открытое ПО или проприетарный продукт);
- популярностью ПО, ограничениями использования в организации;
- требованиями научного журнала, издательства;
- предпочтениями пользователя.

Наибольшую популярность получил пакет Microsoft Office, в частности приложение MS Word. Приложения пакета Office отличаются достаточно низким порогом вхождения в систему и реализуют основные возможности по набору и форматированию текста документа в визуальном редакторе (см. пункт 1.1.3).

С другой стороны, анализ работ показывает, что большинство активных пользователей пакета Office (в том числе работников сферы образования) не владеют приемами структуризации и рубрикации, автоматизации рутинных операций оформления сложных документов, работой со стилями и шаблонами, несмотря на наличие этих функции в MS Word. Отдельной проблемой является учет психологических особенностей восприятия текстовой и графической информации: рядовой пользователь не обладает соответствующим набором компетенций, в результате чего качество создаваемого им учебного материала часто не соответствует должному уровню [46]. Кроме того, простой визуальный интерфейс приложений MS Office вызывает излишнюю концентрацию пользователя на деталях оформления [1,2,16].

В контексте обозначенных проблем возникает необходимость поиска альтернативных средств подготовки электронных и бумажных изданий высокого полиграфического качества [47]. Издательская система LaTeX способна решить эти проблемы, предоставив эффективные средства автоматизации подготовки документа.

Единого мнения о превосходстве какого-либо ПО для верстки документов нет. Выбор инструмента разработки зависит от поставленных задач. Часто специалисты комбинируют доступные средства, соблюдая баланс между качеством документа и трудоемкостью по его разработке.

В 2014 году группа исследователей кафедры психологии, экспериментальной психологии и когнитивных наук университета Гиссена (Германия) провела сравнительное исследование текстового процессора MS Word и издательской системы LaTeX на предмет удобства и эффективности использования при подготовке научных статей [21].

В рамках исследования было задействовано 40 исследователей и аспирантов из шести немецких университетов и по разным научным дисциплинам, которые использовали в своей учебной и научной практике MS Word и LaTeX. Участники были разделены на 4 группы, по 10 участников в каждой: новички в MS Word, эксперты в MS Word, новички LaTeX и эксперты LaTeX.

Испытание заключалось в наборе статьи, включающей блоки непрерывного текста, текст с таблицами и подзаголовками, а также текст с математическими формулами.

Исследование выявило следующие результаты производительности:

1. Пользователи MS Word (за отведенное время) набирают больше непрерывного текста и текста в таблицах, чем пользователи LaTeX.
2. Пользователи LaTeX (в том числе эксперты) допускают больше орфографических и грамматических ошибок, а также ошибок форматирования.
3. По большинству параметров эксперты LaTeX продемонстрировали меньшую производительность, чем новички в MS Word.
4. Новички и эксперты LaTeX в целом набрали больше текста с формулами, чем пользователи MS Word.

По анализу эргономики получены следующие выводы:

1. Пользователи MS Word оценили свое программное обеспечение как менее эффективное, в отличие от пользователей LaTeX.
2. Пользователи LaTeX оценили свою работу как менее утомительную, чем пользователи MS Word.
3. Пользователи LaTeX значительно чаще указывали, что получают удовольствие от использования своего ПО.

Авторы исследования рекомендуют научным учреждениям тщательнее подходить к выбору программного обеспечения. Согласно анализу научных статей организацией American Association for the Advancement of Science, приблизительная оценка доли работ, предоставленных для печати в научную прессу в формате LaTeX (TeX), составляет 26% от общего количества, со значительными вариациями между научными дисциплинами [22].

Анализ работ и отзывов пользователей показывает, что LaTeX эффективен при подготовке следующих типов документов:

- текстов с большим числом формул и математических выражений, текстов естественно-научного цикла (физика, химия, биология);
- текстов научных книг, журналов, статей, где требуется четко заданное форматирование и высокое качество печатного документа;

- текста работ с автоматически генерируемым содержанием, библиографией, перекрестным цитированием.

Пользователи LaTeX отмечают следующие аргументы в его пользу:

- возможность отделить содержание от оформления, эффективно организовать работу с большим числом документов в рамках проекта;
- независимость от операционной системы;
- свободно-распространяемое ПО;
- возможность индивидуального выбора и конфигурации средств верстки документов.

Зависимость между затратой времени (усилий) и сложностью (размером) документа условно можно определить графиком, предложенной М. Пинтериком (рис. 1.23). Он в целом подтверждает практику использования MS Word и LaTeX: для небольших документов с простым форматированием существенное преимущество у MS Word; с ростом объема документа повышается трудность и время форматирования, балансировки содержимого, в то время как LaTeX делает это в автоматическом режиме.

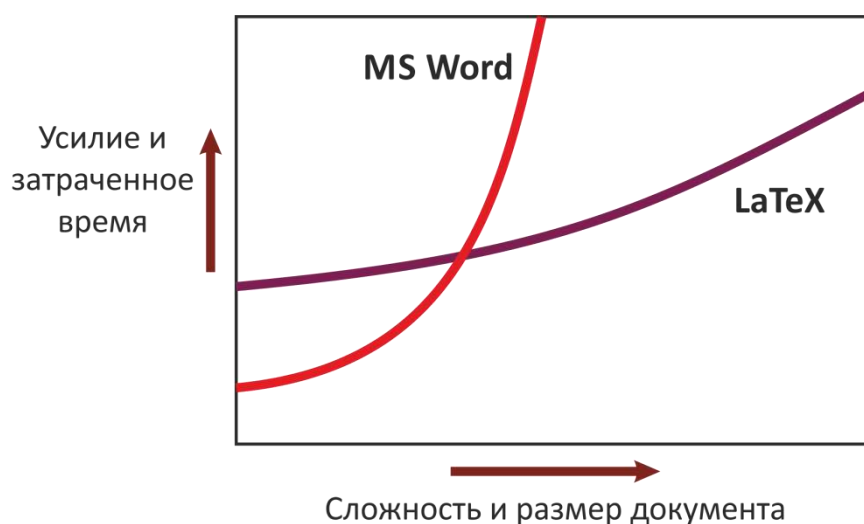


Рис. 1.23. Зависимость между сложностью и усилиями при подготовке документов в LaTeX и MS Word.

Также в пользу LaTeX отметим и скорость обработки больших документов, содержащих многочисленные иллюстрации: в отличие от офисных пакетов, система хранит изображения в отдельных файлах, что позволяет ускорить процесс компиляции документа.

1.5.2 LaTeX в научной работе

Основная ниша использования LaTeX – научная отрасль. Изначально LaTeX создавался для подготовки технических и математических работ, и даже в настоящее время качество подготовки математических текстов в LaTeX остается непревзойденным.

Однако возможности LaTeX не ограничиваются точными, техническим или естественными науками. За время существования системы LaTeX разработано большое число дополнительных пакетов (классов) для оформления документов разной степени сложности и направленности. Сегодня LaTeX де-факто является стандартом публикации научных документов [33].

LaTeX используют для подготовки научных статей, монографий, докладов конференций, тезисов, академических журналов, учебных книг, методических материалов курса лекций и практических занятий, презентационных материалов и постеров. Формат TEX используется такими крупными издательствами, как Elsevier, Springer Science + Business Media, John Wiley & Sons.

Спектр документов, подготовка которых может быть автоматизирована LaTeX, очень широк. Например, пользователь сможет найти различные шаблоны для оформления деловых писем (рис. 1.24) и научных статей для международных журналов (рис. 1.25).

LaTeX выгодно отличается от других типов настольных издательских систем единообразием оформления текста, формул, таблиц. Система автоматизирует сборку разных компонент документа, учитывая стандарты компоновки и балансировки, чтобы в итоге пользователь получил ожидаемый результат не вдаваясь в детали оформления.

Кроме того, система «дисциплинирует» автора к оформлению рукописи в определенном формате и по шаблону, что позволяет избежать проблем при компоновке разных работ в одну. LaTeX дает автору профессиональные возможности издательской системы, но не перегружает его деталями, важными для оформления: эту часть работы он берет на себя.



UNIVERSITY
OF CALIFORNIA

John Smith, PhD

John Smith
Department of Mathematics
123 Broadway
Berkeley CA 12345
Phone: (000) 111-1111
E-mail: j.smith@berkeley.edu
URL: <http://www.johnsmith.com>

February 8, 2014

Prof. Jones
Mathematics Search Committee
Department of Mathematics
University of California
Berkeley, California 12345

Dear Sir or Madam,

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent a nisi diam. Morbi consequat facilisis mi, sit amet laoreet velit aliquet quis. Sed a nisi vel augue ultricies blandit. Phasellus et congue dolor, at cursus dui. Integer quis faucibus metus. Vestibulum lobortis ligula in lectus pretium, in placerat lacus iaculis. Mauris nibh erat, condimentum at tortor at, sagittis viverra tortor. Vivamus posuere fermentum eros, rhoncus sagittis nisi imperdiet ac.

Pellentesque hendrerit neque quis quam fringilla, vitae vulputate quam bibendum. Fusce in hendrerit mauris. Mauris pretium libero eget convallis mattis. Vivamus nec nisi imperdiet, lacinia diam id, facilisis nulla. Vivamus eleifend augue ut libero tincidunt commodo. Vivamus sodales in lacus vitae dictum. Nam et semper felis. Integer scelerisque accumsan condimentum. Aliquam laoreet erat vitae ornare consequat. Donec enim lacus, rutrum ut dui a, pretium mattis dui. Vivamus vulputate arcu nec congue convallis. Suspendisse faucibus turpis ac neque gravida, vel aliquet mauris tristique. In auctor fringilla nunc, sit amet tristique diam semper ac. Cras egestas nisi eu turpis facilisis, ac scelerisque elit lacinia. Sed eget adipiscing enim. Ut pulvinar ultrices purus ac eleifend.

Proin nunc erat, vestibulum quis tincidunt sit amet, cursus et libero. Aliquam erat volutpat. Sed vel malesuada velit. Vivamus risus arcu, facilisis non mauris sed, interdum varius tellus. Vivamus aliquam vel sapien quis ultricies. Mauris venenatis risus enim, in dapibus quam volutpat in. Vivamus iaculis sapien sit amet massa vehicula, tempus facilisis nulla adipiscing. Mauris ullamcorper pharetra libero, quis sodales dolor blandit et. Donec convallis risus id euismod pharetra. Suspendisse placerat laoreet imperdiet. Praesent malesuada fringilla urna, non sollicitudin erat. Nullam arcu neque, lobortis nec facilisis at, scelerisque ac erat. Nunc quis tincidunt lorem, vitae laoreet enim.

Sincerely,

John Smith, PhD

Рис. 1.24. Модельный пример делового письма, набранного в LaTeX.

В. В. Жиков
Владимирский государственный
университет им. А.Г. и Н.Г. Столетовых
Владимир, Россия

Д. А. Якубович
Владимирский государственный
университет им. А.Г. и Н.Г. Столетовых
Владимир, Россия

ГАЛЕРКИНСКИЕ ПРИБЛИЖЕНИЯ В ЗАДАЧАХ С p -ЛАПЛАСИАНОМ

Изучается эллиптическая задача с p -лапласианом и строится система галеркинских приближений. Оценивается разность между точным и приближенным решением для случая постоянного и переменного показателя p . Библиография: 3 назв.

1. Оценка для постоянного показателя p

Рассмотрим задачу Дирихле

$$-\operatorname{div}(|\nabla u|^{p-2}\nabla u) = f, \quad u|_{\partial\Omega} = 0, \quad (1.1)$$

в ограниченной гладкой области $\Omega \subset \mathbb{R}^d$, где $p > 1$. Функция u принадлежит пространству Соболева $X = W_0^{1,p}$, f — линейный непрерывный функционал на пространстве X , т.е. $f \in X^*$. Левая часть (1.1) называется p -лапласианом и обозначается $\Delta_p u$. Норму в пространстве X определим как

$$\|u\|_X = \|\nabla u\|_p = \left(\int_{\Omega} |\nabla u|^p dx \right)^{\frac{1}{p}}.$$

Определение 1.1. Под решением задачи (1.1) понимается функция $u \in X$ такая, что

$$\int_{\Omega} |\nabla u|^{p-2} \nabla u \cdot \nabla \varphi dx = (f, \varphi) \quad \forall \varphi \in X. \quad (1.2)$$

Известно, что задача (1.2) — это уравнение Эйлера вариационной задачи

$$\min_{u \in X} \int_{\Omega} \left(\frac{|\nabla u|^p}{p} - g \cdot u \right) dx,$$

решение которой существует и единственно.

Работа выполнена при поддержке Российского фонда фундаментальных исследований (проект №. 14-01-00192 А).

© В. В. Жиков, Д. А. Якубович, 2016

Рис. 1.25. Пример научной статьи для математического сборника, набранной средствами LaTeX.

Базовые возможности работы с математической символикой встроены в стандартный набор классов LaTeX. Многие известные издания математической научной литературы следуют стандартам американского математического общества (AMS), в связи с чем был разработан ряд AMS-пакетов для оформления статей, например:

- **amsmath** – основной стиль AMS-LaTeX с дополнительными окружениями для оформления формул;
- **amssymb** – пакет для набора математических символов;
- **amsthm** – оформляет теоремоподобные структуры.

Среди пакетов для оформления графиков и диаграмм выделяют пакеты:

- **pgfplots** – мощный пакет, предоставляющий инструменты для создания двумерных и трехмерных графиков и диаграмм с возможностью их настройки;
- **tikz** – содержит инструменты по созданию схем и диаграмм.
- **PSTricks** – реализует инструменты по двумерным графиков, позволяет строить математические функции Фурье, распределения Гаусса и прочее;
- **tikz-3dplot** – рисует 3D базис, вектора и поверхности.

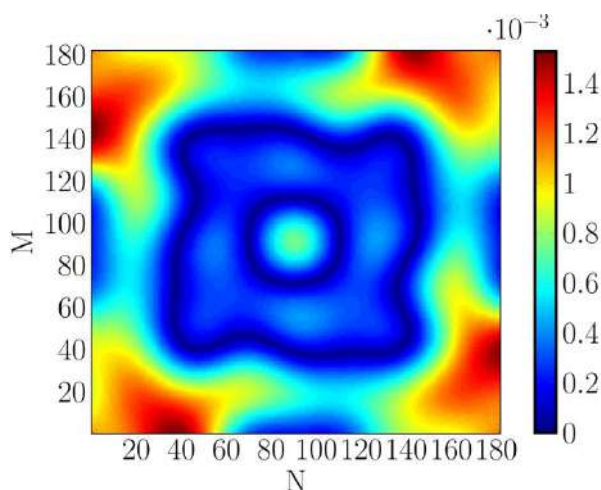


Рис. 1.26. График, построенный с помощью пакета pgfplots.

Стандартные классы **book**, **report**, **article** и **letter** хорошо подходят для верстки научных работ и сопровождающих документов. Однако в последнее время развитие этих пакетов остановлено, наблюдается отставание от современных стандартов оформления.

Для решения этой проблемы используются усовершенствованные пакеты, например:

- классы **KOMA-Script** являются отличной альтернативой стандартным классам; функциональность пакета обеспечивается классами **scrartcl** (статья), **scrreprt** (отчёт), **scrbook** (книга) и **scrlttr2** (письмо);
- **memoir** – класс для создания простых книг и математических публикаций; имеет подробную документацию;
- **beamer** – современный и активно развивающийся пакет для подготовки электронных презентаций, содержит обширный набор встроенных стилей и шаблонов (рис. 1.27).

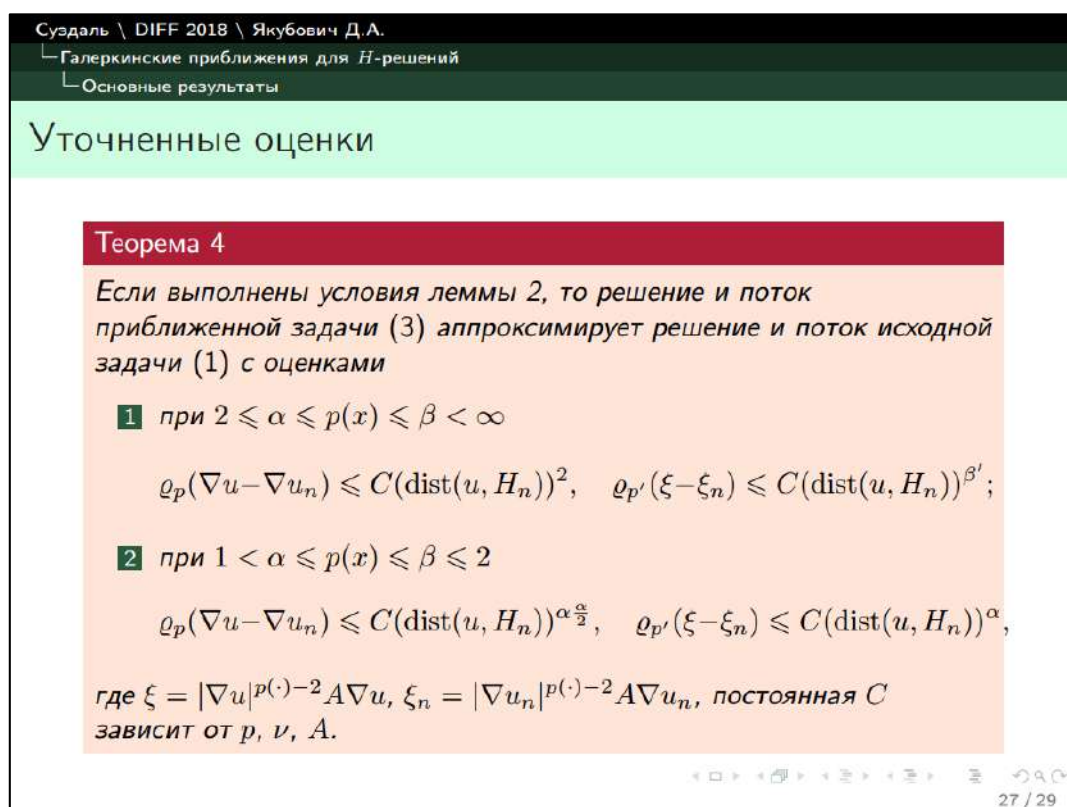


Рис. 1.27. Класс Beamer используется для подготовки высоко качественных презентаций.

Для русскоязычной аудитории научных работников полезным может оказаться класс **disser**, предназначенный для верстки работ на степень бакалавра, магистра, кандидата и доктора наук.

Более полный обзор дополнительных пакетов CTAN для LaTeX проведен Балдиным Е. в цикле статей IBM Developers [26].

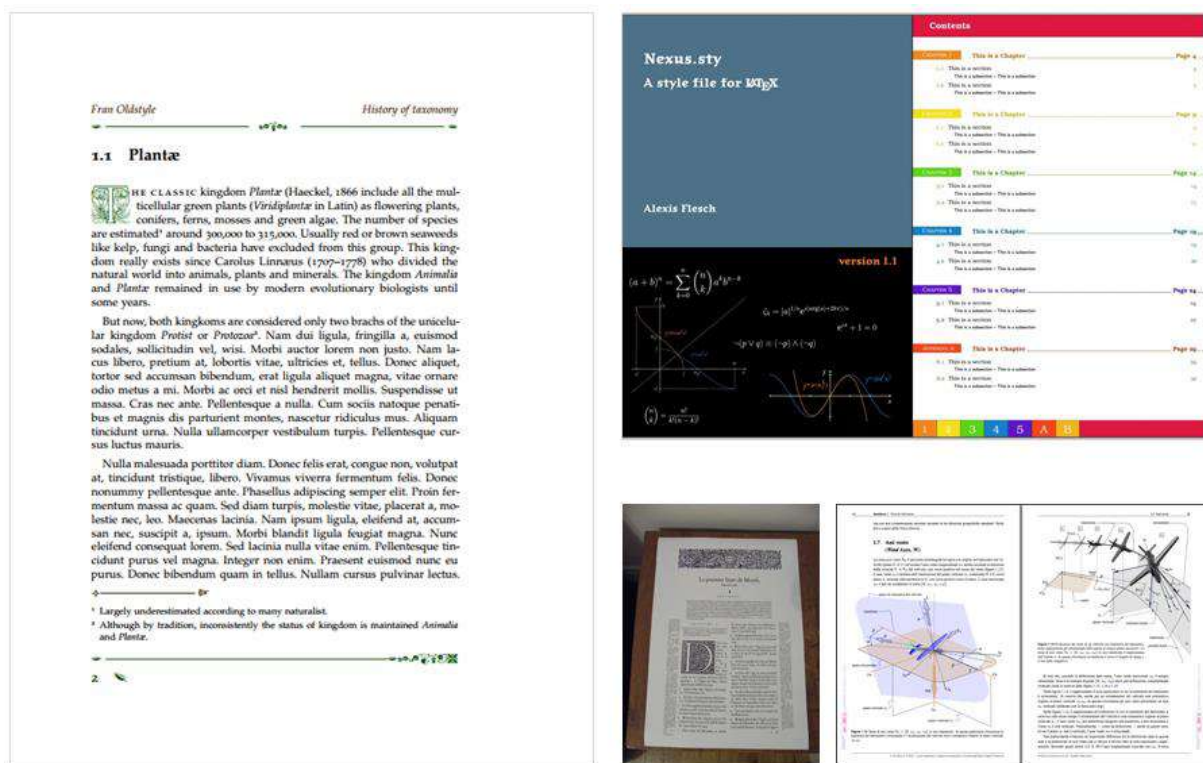


Рис. 1.28. Профессиональные пользователи LaTeX превращают систему в мощное средство разработки документов.

1.5.3 LaTeX в практике учителя

При подготовке текущего учебного курса был проведен анализ материалов сети Интернет, ряда существующих авторских методик обучения работе с LaTeX. Установлено, что:

1. практика использования издательской системы LaTeX учителями математики, информатики, физики, химии крайне мала;
2. учителя гуманитарного направления практически не используют систему LaTeX (доля существующих работ и публикаций относительно общего потока настолько мала, что ей можно пренебречь);
3. основная доля найденных работ в формате TEX являются иностранными.

Основные причины непопулярности системы LaTeX в работе учителей и преподавателей:

1. Повышенный порог вхождения в систему. В первую очередь, он обусловлен концепцией логической разметки до-

кумента, в определенном смысле противоположной приложениям с визуальным интерфейсом.

2. Дополнительное время на изучение базовых навыков верстки в LaTeX. В отличие от визуального редактора, интуитивное обучение здесь невозможно. Требуется знакомство с специальной учебной литературой.
3. Программное обеспечение. Установка и конфигурация ПО требует предварительного ознакомления.
4. Ограниченность в использовании ПО учебным заведением.
5. Малое число учебных источников, доступно излагающих материал для широкого круга читателей.
6. Отсутствие мотивации самого педагога. Часто предпочтение отдается более простой системе, при этом в ущерб качеству итогового документа.
7. Мало уделяется внимания культуре профессионального подхода в деле верстки учебно-методических изданий.

Последний фактор играет важную роль. Некачественно оформленный текст трудно воспринимается, требует больше времени на обработку и понимание, вызывает преждевременную усталость при чтении, что способствует потере интереса учащегося [2].

Качественной работой педагог демонстрирует приверженность научному знанию, формируя базу для критического мышления учащегося в вопросе самостоятельного выбора учебной литературы, способностью анализа ее качества.

Низкие показатели распространения LaTeX в учебно-образовательной среде Р.Ф. связаны с более низким уровнем поддержки системы русскоязычным сегментом пользователей. Основная причина – LaTeX приобрел популярность в то время, когда визуальные редакторы были менее функциональны и доступны (80-90 годы), а массовая компьютеризация населения Р.Ф. началась с 00х, во время развития WYSIWYG-технологий. Кроме того, на западе жестче регулируются права на использование коммерческих продуктов, поэтому многие пользователи стремятся использовать открытое ПО.

Перечислим основные виды документов, которые могут быть созданы педагогом в LaTeX для использования в учебно-образовательном процессе и научной деятельности:

1. **Материалы занятий** (конспекты уроков, лекций, практических и лабораторных занятий), содержащих большой объем текста. В этом случае команды рубрикации автоматически разбивают документ на разделы, результат оформляется в виде книги, доклада или статьи. Заранее оформленный шаблон допускает многократное применение: меняется только содержимое. Для оформления подобных документов достаточно знать базовые возможности стандартных классов.
2. **Учебное издание с большим числом математических символов и формул.** Требование к высокому качеству их оформления лежало в основе TeX. Набор формул в LaTeX осуществляется с помощью специальных команд и клавиатуры, нет необходимости одновременно использовать манипулятор (мышь) для поиска нужного символа или оператора, как в визуальном редакторе. Для тонкой настройки могут оказаться полезными пакеты серии AMS.
3. **Учебное издание с большим числом перекрестных ссылок.** Документ с развитой навигацией прост и удобен в использовании. LaTeX поддерживает мощный пакет `hyperref`, позволяющий создавать локальные и внешние ссылки на разделы, формулы, библиографические источники, веб-страницы и многое другое.
4. **Презентации, выдержанные в лаконичном академическом стиле.** В этом направлении отлично зарекомендовал себя пакет Beamer.
5. **Дополнительные шаблоны по оформлению тестов, проверочных работ, экзаменационных билетов** и других материалов, ускоряющие набор документа. Для подобных работ CTAN предоставляет, например, пакет `exam`. Профессиональные пользователи LaTeX создают интересные интерактивные формы документа; ряд шаблонов можно найти на ресурсах Overleaf.com и документации CTAN [27].
6. **Научная работа.** Серьезные издательства и научные журналы (прежде всего, по математике и физике) требуют работы именно в формате TEX. Перенабор редакторами журнала текста документа, например, из формата DOC / DOCX станет отдельной статьей расходов для автора.

7. **Любое издание**, требующее качественной проработки и высокого качества печати.
8. Многие **веб-ресурсы** учебного и энциклопедического характера поддерживают частичную работу с LaTeX. Например, ресурс Wikipedia отображает формулы в виде изображений, сгенерированных на базе TeX-разметки.

Вопросы для самопроверки

1. В каких отраслях LaTeX находит широкое применение?
2. Почему LaTeX мало популярен в среде учителей?
3. Перечислите возможности, которые дает LaTeX учителю.
4. В каких вопросах LaTeX может быть полезен преподавателю и научному работнику?
5. Что такое CTAN и каково его основное назначение?

Практикум

1. В сети Интернет найдите пример(ы) любой научной статьи, выполненной в LaTeX. Охарактеризуйте качество ее оформления.
2. В сети Интернет по ключевому слову «Beamer» найдите примеры презентаций. Какие интересные возможности для себя вы увидели?
3. Почему качество оформления учебного издания влияет на восприятие, запоминание и понимание учащихся? Обоснуйте ответ, ссылаясь на известные вам работы психологов и педагогов.
4. Укажите качества, которые может сформировать у специалиста сферы образования работа с системой LaTeX.
5. Чем полученные в работе с LaTeX компетенции могут помочь, например, при работе с другим офисным пакетом?
6. Изучите практику использования LaTeX в вашем учебном заведении. Какие отделы (кафедры) его используют?

1.6 Практика использования LaTeX авторами

1.6.1 Апробация материалов

Ранее мы уже отмечали, что несмотря на востребованность системы LaTeX в научной сфере, ее изучение, как правило, не включено в программу подготовки бакалавров, специалистов и магистров даже естественно-научного направления.

С учетом обозначенных возможностей нами было принято решение расширить программу подготовки студентов кафедры математического образования и информационных технологий Педагогического института ВлГУ, включив в одну из дисциплин работу с LaTeX. Главная цель – расширить представление студентов о возможностях настольных инструментов разработки документов на базе издательской системы LaTeX.

Апробация учебных материалов курса проходила в два этапа.

На первом этапе (осенний семестр 2017-2018 учебного года) учебные материалы по LaTeX были предложены студентам кафедры социальной педагогики и психологии Педагогического института ВлГУ. В рамках самостоятельной работы по дисциплине «Информационные технологии в гуманитарных и социальных науках» учащиеся познакомились с основами LaTeX: созданием документа, командами форматирования текста, работой с рубриками, изображениями и таблицами, перекрёстными ссылками. В качестве учебного материала использовалась разработанная авторами текущего пособия учебная презентация (см. рис. 1.29).

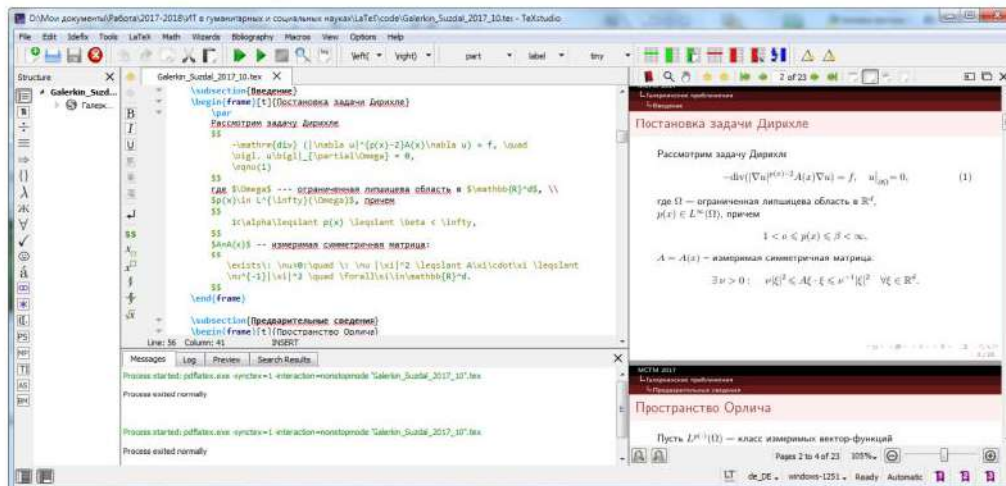
На втором этапе (весенний семестр 2018-2019 учебного года) апробация уже проходила на кафедре математического образования и информационных технологий Педагогического института ВлГУ в рамках учебной дисциплины «Информационные технологии в научной деятельности». На работу было отведено 5 лабораторных занятий и соответствующие учебному плану часы самостоятельной работы. В отличие от первого этапа, материал был расширен дополнительными темами: работа с формулами, создание новых команд, разработка презентаций.

Программное обеспечение для работы с LaTeX

Что нужно разработчику?

LaTeX – это набор специальных пакетов (библиотек классов), в которых реализован функционал для верстальщика. Он не является отдельной программой, в которой документ можно создать и, например, просмотреть, распечатать или сохранить.

Для работы с LaTeX нужно специальное ПО, которое позволит оперировать возможностями его библиотек, и желательно, чтобы это было удобно.



Якубович Д. А., Еропова Е. С., Владимир 2017

Генерация оглавления

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}

% Автоматическая генерация оглавления
\tableofcontents

\section[Секция 1]{Название секции 1}
\subsection*{Введение}
\addcontentsline{toc}{subsection}{Введение}
...
\subsection{Подзаголовок 1}
...
\subsection{Подзаголовок 2}
...
\subsection{Подзаголовок 3}
...
\subsection*{Заключение}
\addcontentsline{toc}{subsection}{Заключение}
...

\end{document}
```

Теперь *-рубрики
включаются в
оглавление

Содержание

1 Секция 1	1
Введение	1
1.1 Подзаголовок 1	1
1.2 Подзаголовок 2	1
1.3 Подзаголовок 3	1
Заключение	1

1 Название секции 1

Введение

...

1.1 Подзаголовок 1

...

1.2 Подзаголовок 2

...

1.3 Подзаголовок 3

...

Заключение

...

Якубович Д. А., Еропова Е. С., Владимир 2017

Рис. 1.29. Пример слайдов из учебной презентации, использованной для апробации курса по LaTeX.

Цель проводимых апробаций – выявить наиболее трудные вопросы в изучении системы LaTeX и источники часто возникающих у студентов ошибок.

Анализ выполненных работ и отзывов студентов показал следующее:

- учащиеся редко сталкивались с проблемами набора и форматирования обычного текста и текста, содержащего математические формулы, частые ошибки возникали при верстке таблиц и изображений, перекрестных ссылок, локальном нестандартном форматировании текста;
- скорость набора любого типа текста в целом оказалась ниже, чем при работе учащихся с пакетом MS Office, однако при форматировании большого объема скопированного текста отмечался прирост производительности;
- учащиеся отметили высокую эффективность и удобство автоматической рубрикации, генерации оглавления и механизм работы перекрестных ссылок;
- подавляющее большинство испытуемых отметили высокое качество оформления математических формул и гармоничное оформление документа в целом;
- среди основного недостатка учащиеся указывали на достаточно трудоемкий процесс настройки нестандартного оформления, переопределения базовых параметров и работу с дополнительными пакетами.

В результате апробации были внесены следующие изменения в структуру, содержимое и реализацию курса:

1. более подробно затронута тема сравнения концепций WYSIWYG и логической разметки документов;
2. материалы расширены примерами, которые подробно прокомментированы;
3. интерфейс курса реализован в форме веб-учебника, который предоставляет удобный функционал для учащегося и сосредотачивает в себе все учебно-методическое обеспечение курса.

По результатам апробации учебный курс был внедрен в работу кафедры математического образования и информационных техноло-

гий ВлГУ в рамках дисциплины «Информационные технологии в научной деятельности».

1.6.2 Электронный учебник

Для удобства работы с учебным материалом был разработан веб-учебник. Реализация содержания в форме электронного учебника имеет следующие достоинства:

- удобная форма представления информации;
- эргономичный интерфейс (наличие развитой системы навигации);
- возможность работы с отдельными файлами;
- единый каталог всего проекта;
- работа с курсом на разных устройствах.

Мы не выбирали какое-либо готовое решение, шаблон, приложение для разработки электронного учебника. Электронное учебное пособие реализовано согласно предложенной авторами структуре и соответствует необходимым требованиям к оформлению электронных учебных ресурсов.

Использование технологий HTML5, CSS3 и JavaScript позволило:

- реализовать курс в удобной для работы форме – в виде веб-сайта (для работы с учебным материалом достаточно наличия у пользователя браузера);
- работать с мультимедиа: изображения, эффекты анимации;
- построить структуру учебника, оптимизировав ее под содержание учебного курса;
- избавиться от ограничений в оформлении и дизайне страниц курса, возникающих при использовании других инструментальных средств;
- реализовать современный адаптивный дизайн для стационарных компьютеров, планшетов и смартфонов;
- осуществить функциональную обработку данных пользователя (интерактивный итоговый тест по курсу).

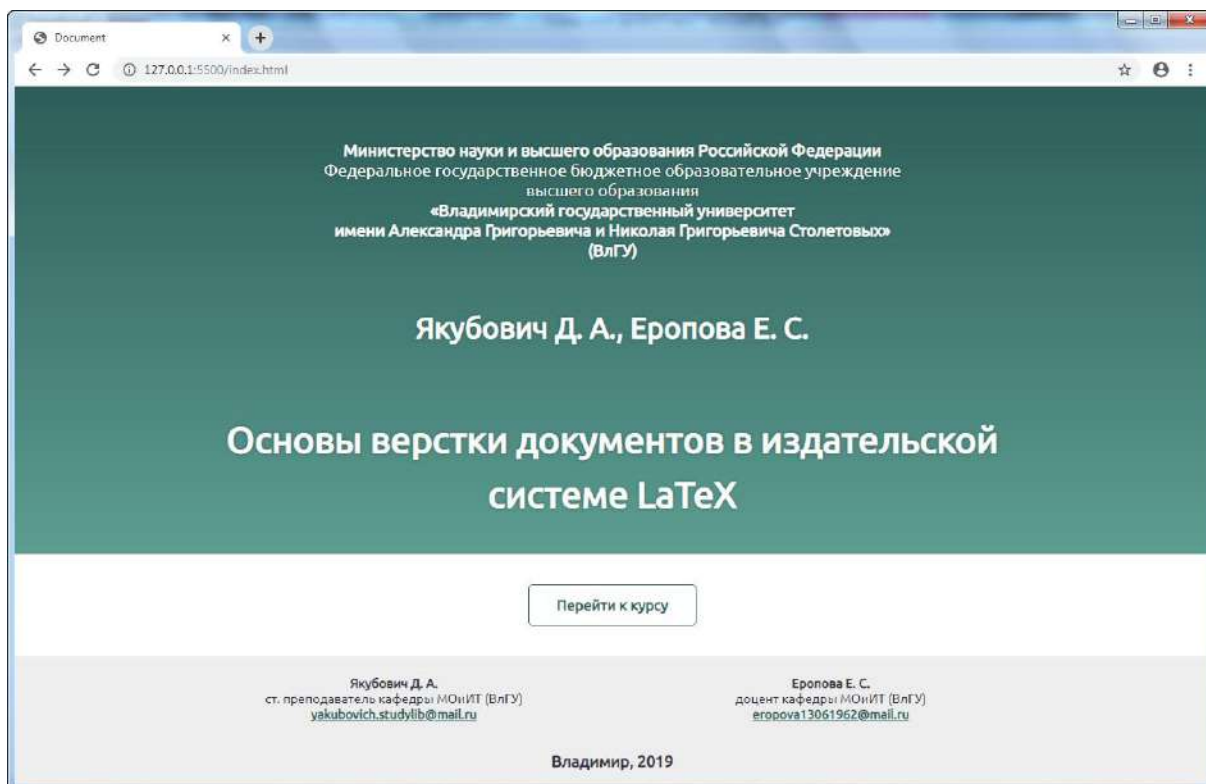


Рис. 1.30. Главная страница учебника.

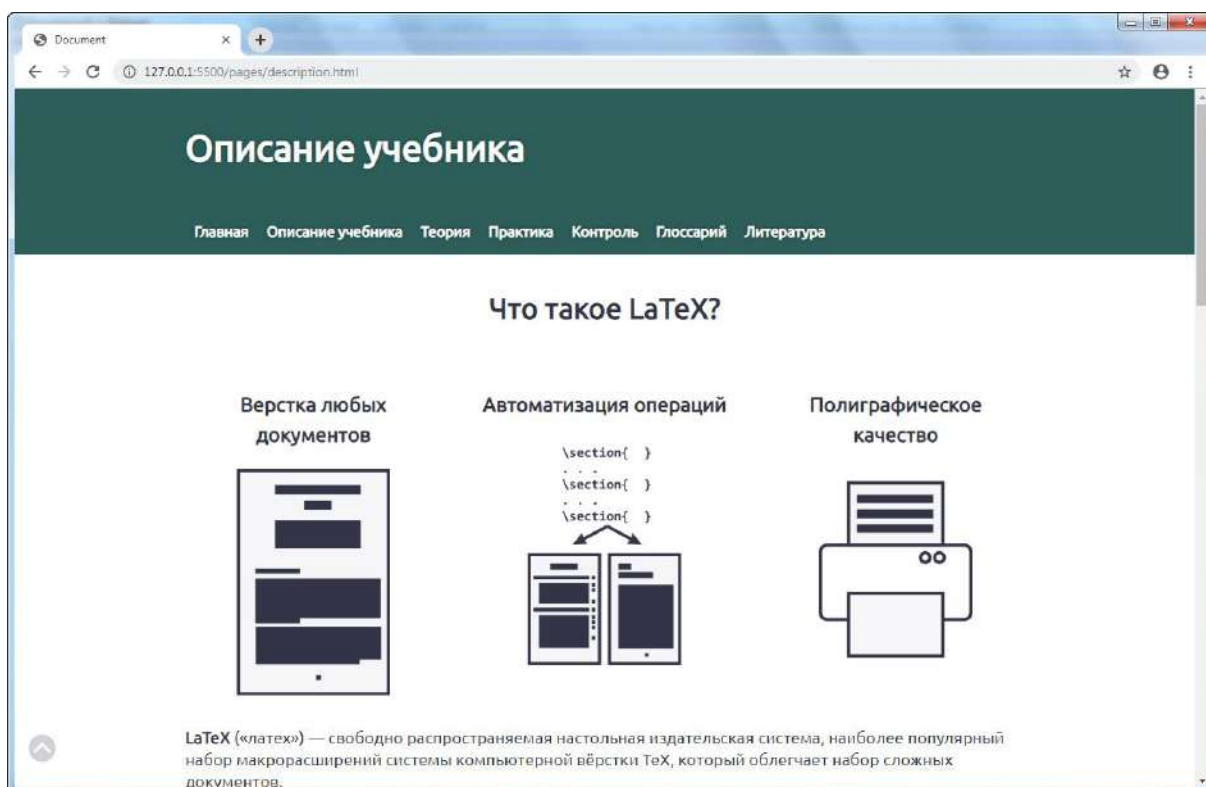


Рис. 1.31. Страница с описанием курса.

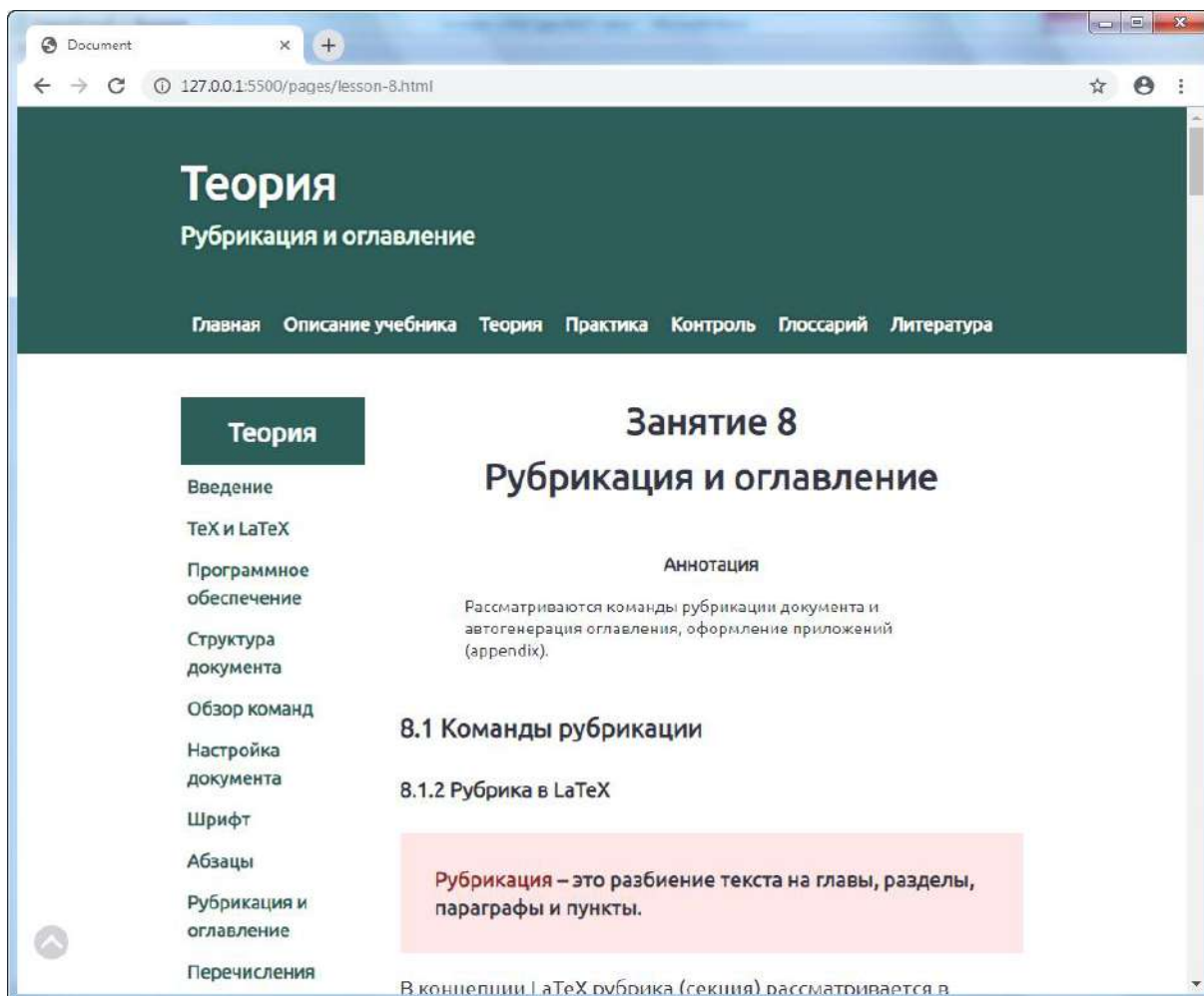


Рис. 1.32. Пример страницы с учебным материалом.

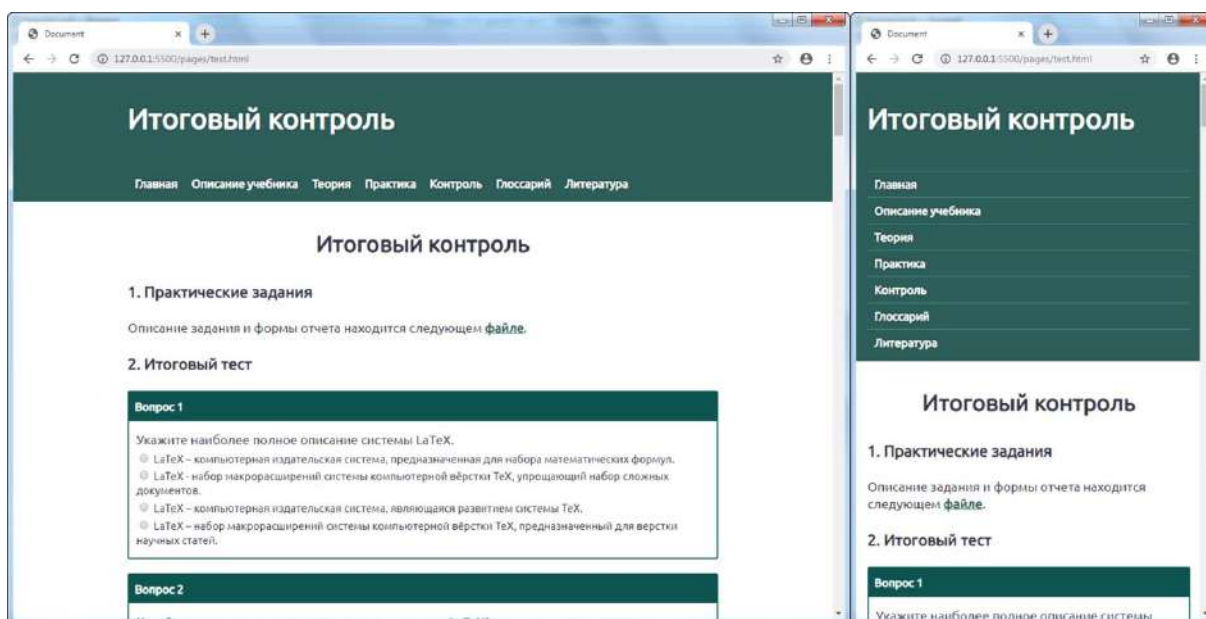


Рис. 1.33. Веб-учебник адаптивен для разных типов устройств.

Кроме перечисленных достоинств, электронное учебное пособие можно использовать для организации как классического очного, так и дистанционного обучения. Курс можно использовать в двух режимах:

1. размещение файлов веб-учебника на локальном компьютере или локальной сети компьютерной аудитории; также файлы изначально могут быть скопированы для дальнейшего использования;
2. размещение на электронном ресурсе сети Интернет; для корректной работы требуется приобрести услуги хостинга или воспользоваться иной возможностью размещения веб-материалов (например, выложить курс на сайте учебного заведения).

При разработке пособия использовалась технология блочной адаптивной верстки. Интерфейс выполнен в лаконичном стиле и с соблюдением особенности цветового восприятия.

Текущее учебное пособие построено на материалах разработанного авторами веб-учебника по LaTeX.

1.7 Программное обеспечение

1.7.1 Основные инструменты разработчика

LaTeX является набором пакетов – специальных библиотек, в которых реализован функционал для верстальщика. Он не является отдельной программой, в которой документ можно создать и, например, просмотреть, распечатать или сохранить. Для работы с LaTeX нужно специальное ПО, которое позволит оперировать возможностями его библиотек, и желательно, чтобы это было удобно.

Стандартный набор для работы с LaTeX включает следующие компоненты:

1. дистрибутив (ПО) в виде пакетов TeX / LaTeX;
2. редакторы кода;
3. средства просмотра документа;
4. компиляторы.

1.7.2 Дистрибутив

К настоящему времени разработано множество пакетов для работы с файлами формата TEX. Несмотря на то, что LaTeX является наиболее популярным, есть и другие пакеты, реализующие дополнительный функционал и возможности по разработке документов (например, XeTeX, PDFLaTeX, TeXify).

Подобные пакеты также включаются в общую сборку, либо могут быть подключены отдельно самим пользователем по мере необходимости.

Среди наиболее известных и полных сборок выделяют TeX Live, MiKTeX и MacTeX. Перечисленные пакеты распространяются свободно.

TeX Live

TeX Live – наиболее полный дистрибутив LaTeX. TeX Live позволяет запускать и устанавливать LaTeX на различных операционных системах, например: Linux, Mac OS X и Microsoft Windows [48].

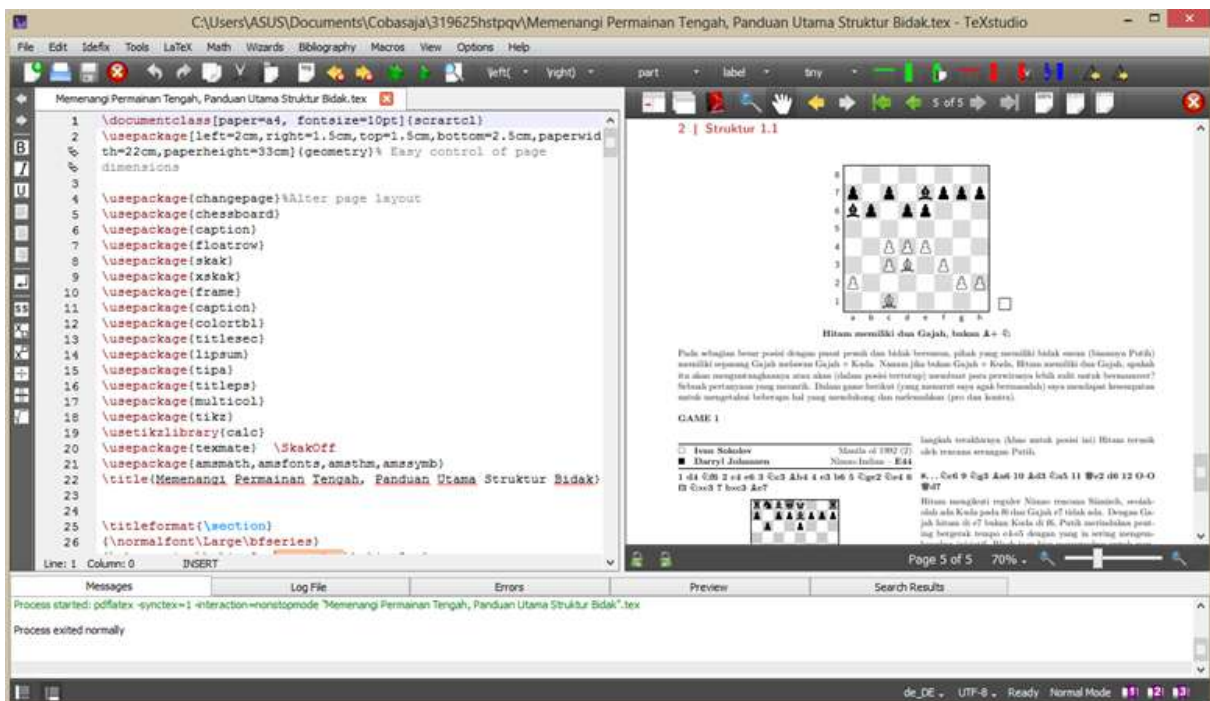


Рис. 1.34. TeX Live в редакторе TeXstudio.

МiKTeX

МiKTeX – открытый дистрибутив TeX для платформы Windows. Поддерживает множество других макропакетов TeX'a. Содержит встроенные редактор TeXworks, что позволяет работать сразу после установки [49].

В настоящее время в состав МiKTeX включены основные компиляторы и библиотеки LaTeX. Дистрибутив поддерживается последними версиями Windows.

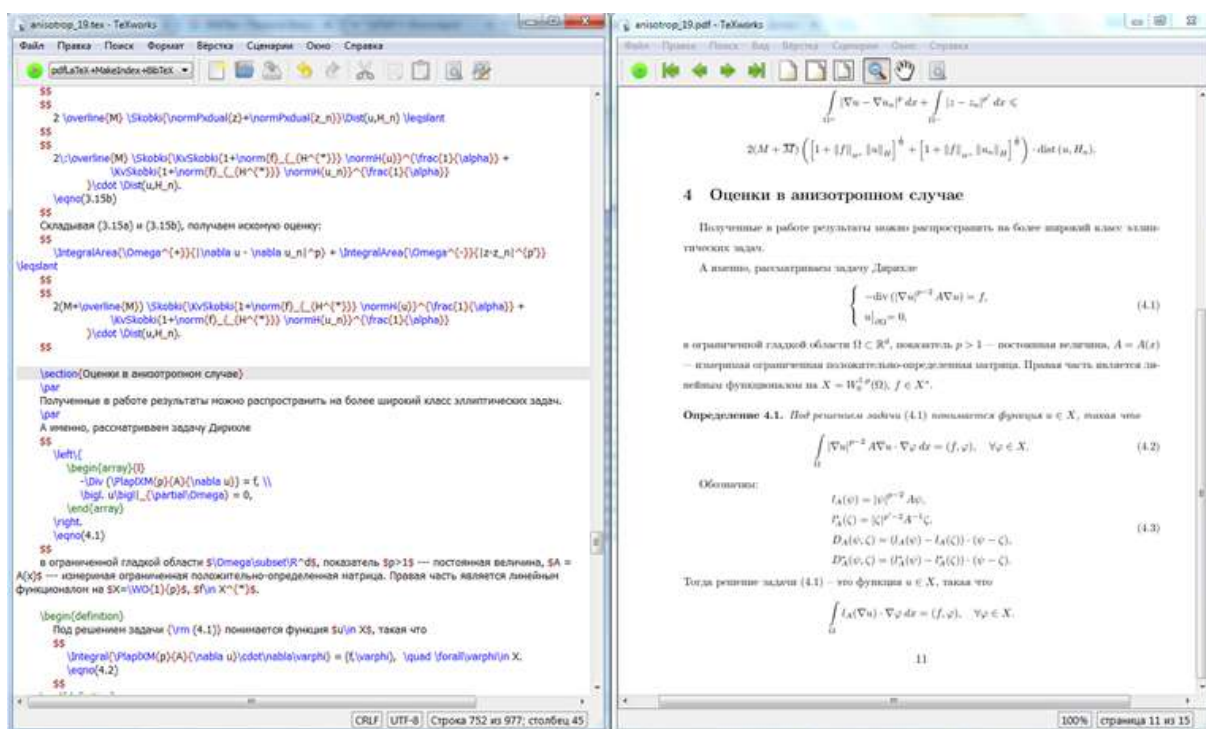


Рис. 1.35. МiKTeX и встроенный редактор TeXworks.

MacTeX

MacTeX – это переиздание TeX Live. Включает в себя утилиты и интерфейсы для Mac [50].

По состоянию на 2019 г содержимое пакета установки MacTeX превышает 4 ГБ. Однако существует и «облегченная» версия – BasicTeX.

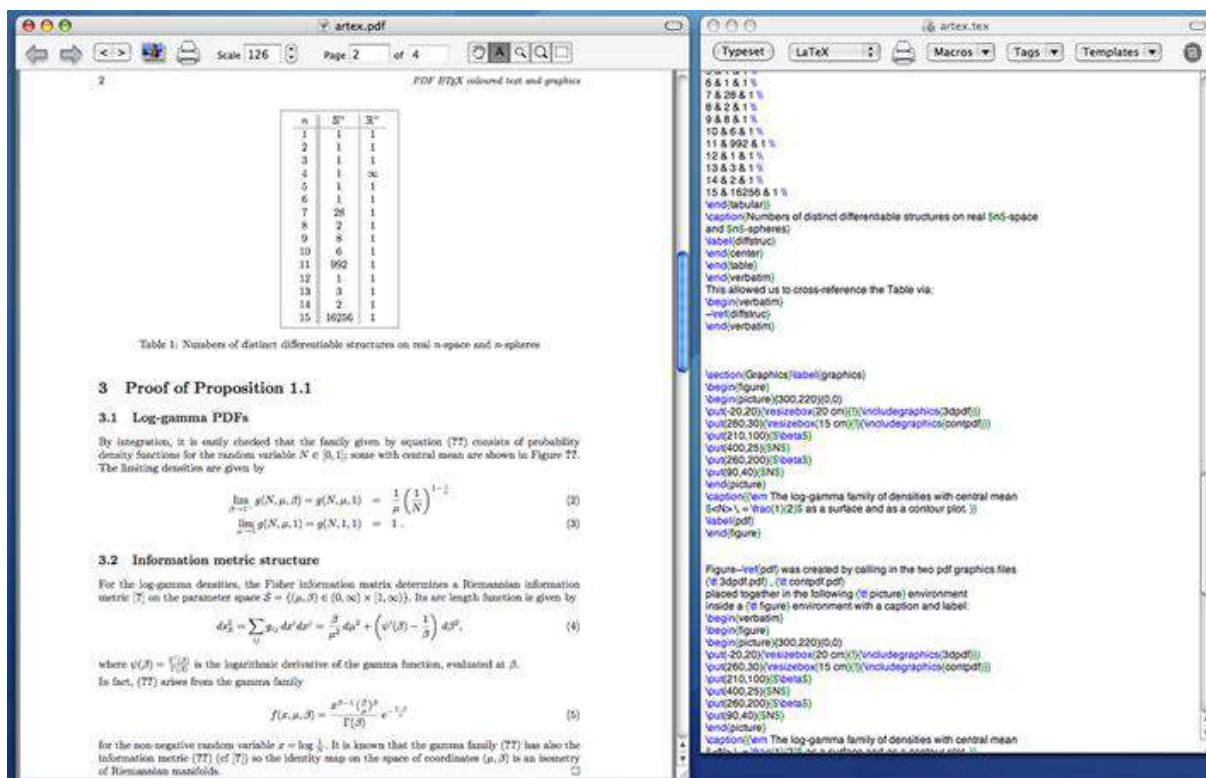


Рис. 1.36. Интерфейс MacTeX.

1.7.3 Редакторы кода

Поскольку основной документ LaTeX – это обычный текстовый файл (в формате TEX), то его можно редактировать в любом текстовом редакторе. Однако стоит отдать предпочтение развитым текстовым редакторам, которые способны подсвечивать синтаксис команд, возможные ошибки, функции расширенного поиска и замены, сборки документа из нескольких файлов и др.

TeXstudio

TeXstudio – свободный кросс-платформенный редактор LaTeX [51].

TeXstudio можно рассматривать как полноценную интегрированную среду разработки документа на базе LaTeX: она поддерживает проверку правописания, подсветку синтаксиса, визуальные компоненты.

В качестве дистрибутива TeXstudio может использовать, например, MiKTeX.

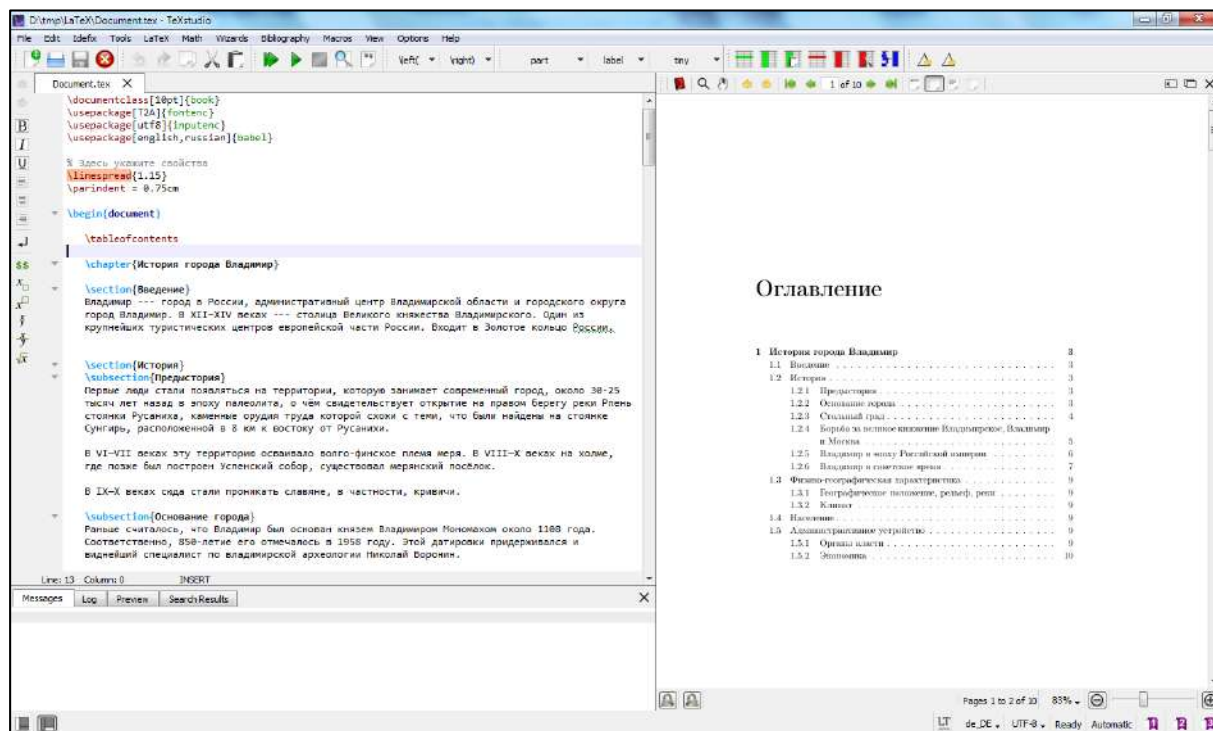


Рис. 1.37. TeXstudio позволяет редактировать код разметки документа и параллельно просматривать конечный результат.

WinEdt

WinEdt — редактор для файлов ASCII и оболочка для MS Windows. Мощный редактор с подсветкой синтаксиса и визуальными шаблонами часто используемых команд [52].

Редактор является условно платным. Однако он отличается большим набором полезных функций и развитым механизмом сборки документов. Программа изначально позиционируется как удобный визуальный редактор для большого числа TeX-систем, таких как MiKTeX, TeX Live и т.д.

WinEdt поддерживает процесс обработки DVI- и PDF-файлов.

На данный момент актуальной версией программы является WinEdt 10.

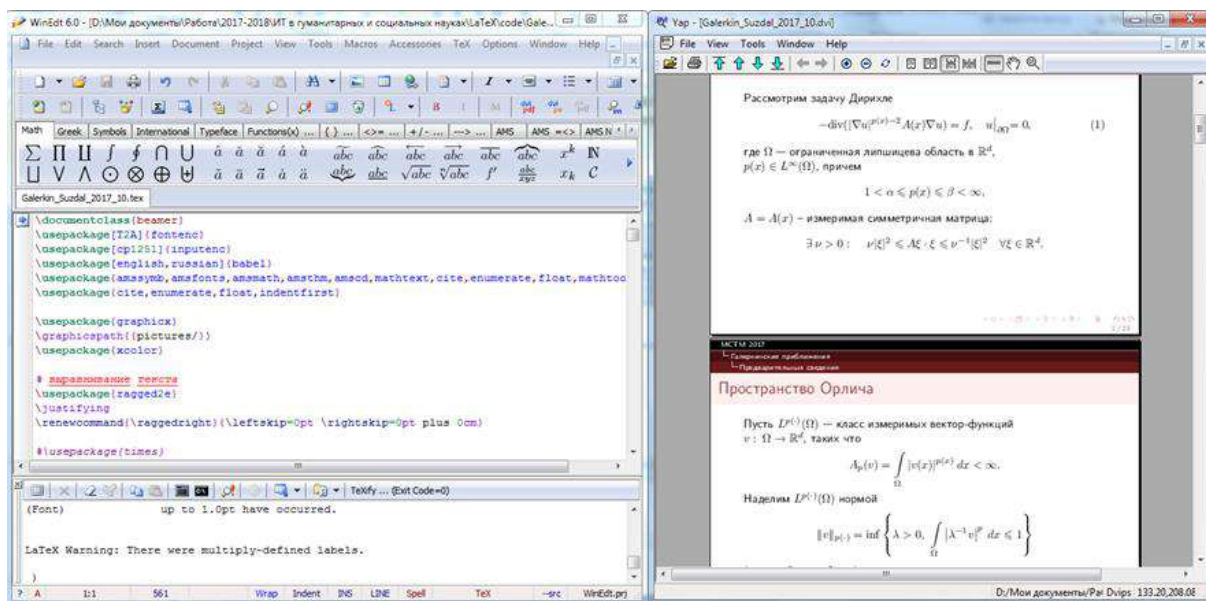


Рис. 1.38. Интерфейс WinEdt.

TeXworks

TeXworks — это свободная среда для работы с TeX-документами, включающая редактор, просмотрщик PDF.

TeXworks требует предустановленного TeX Live, MiKTeX или proTeXt. В настоящее время включается в пакет MiKTeX в качестве редактора по умолчанию (см. рис. 1.35).

1.7.4 Средства просмотра

К средствам просмотра относятся программы, которые позволяют просматривать конечный вид документа, например, в форматах DVI и PostScript.

Как было отмечено ранее, сейчас чаще всего более приемлемым является использование формата PDF. Тем более, что его можно открыть как с помощью множества приложений, так и браузера. Обычно в него и экспортируется окончательный вариант работы.

Одним из наиболее популярных приложений для просмотра DVI является приложение YAP [5]:

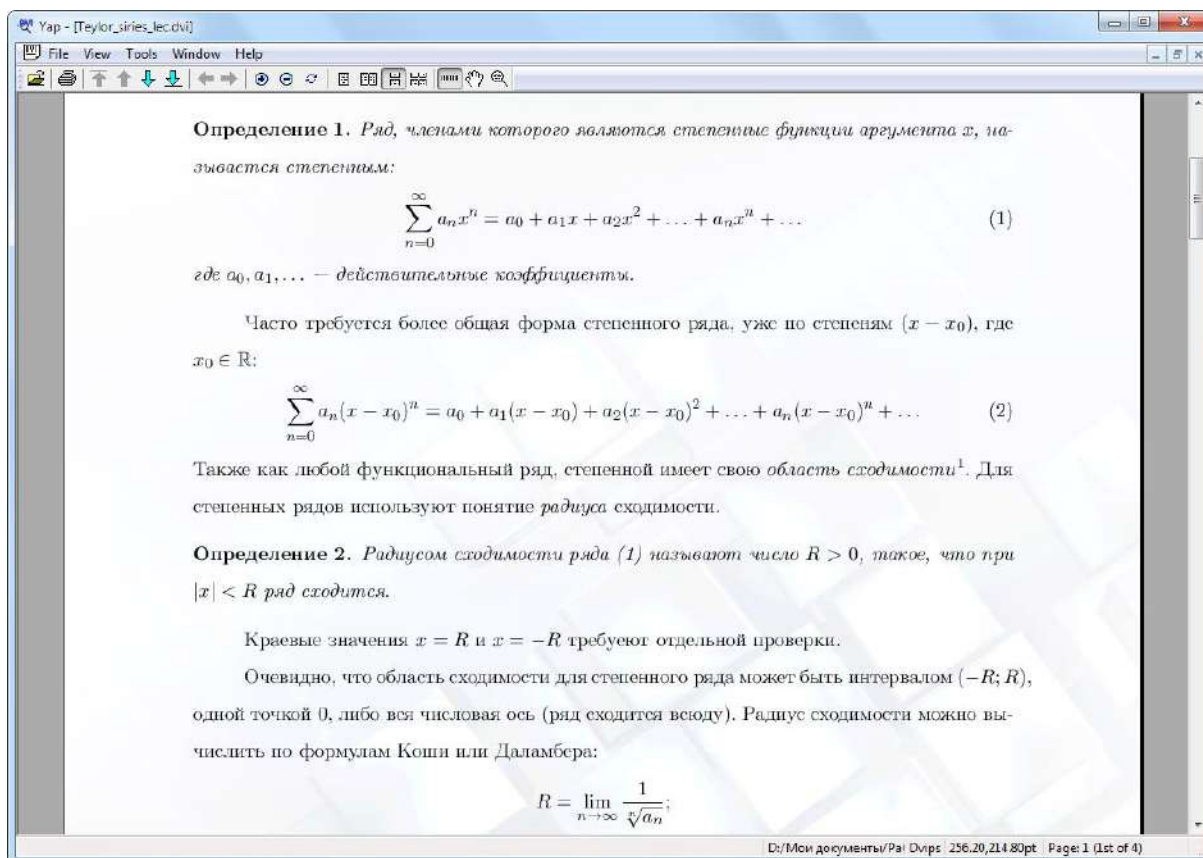


Рис. 1.39. YAP – одна из программ для просмотра DVI-файлов.

Средства просмотра могут быть изначально встроены в редактор кода, что позволяет избегать самостоятельной их настройки.

1.7.5 Компиляторы

Основная задача программы-компилятора – преобразовывать текстовую разметку документа на язык, понятный компьютеру; осуществлять работу с библиотеками дистрибутива. Компиляторы входят в состав дистрибутива.

Работать с компилятором можно «вручную», указывая ему файлы для компиляции и дополнительные опции. Эта процедура аналогична той, что проделывают программисты, осуществляющие сборку программы из разных файлов. Однако рядовому пользователю знать такие тонкости работы не нужно: перечисленные ранее текстовые редакторы осуществляют работу с компилятором автоматически [5].

В зависимости от дистрибутива, указанные (и не только) выше компоненты могут быть «вшиты» в него изначально! Это позволяет лишь установить дистрибутив, задать базовые настройки и сразу же приступить к работе. Опытный пользователь может собирать среду разработки согласно своим предпочтениям.

1.7.6 Веб-сервисы для работы с LaTeX

Для использования LaTeX необязательно устанавливать специальное программное обеспечение.

Веб-сервисы полезны начинающему пользователю, поскольку избавляют его от необходимости изучения тонкостей работы с LaTeX уже в самом начале. А для опытного разработчика это возможность работы с LaTeX в любой точке мира и на любом компьютере, который имеет доступ в сеть Интернет.

В настоящий момент существует несколько крупных веб-сервисов, организующих онлайн-работу с LaTeX:

- Overleaf.com (<https://www.overleaf.com/>);
- ShareLaTeX.com (<https://www.sharelatex.com/>);
- Latexbase.com (<https://latexbase.com/>).

Отметим, что за время написания этого курса проект ShareLaTeX стал частью Overleaf. Поэтому остановим свое внимание на сервисе Overleaf.com.

Overleaf.com – это онлайн сервис для написания и публикации документов на LaTeX [28].

Проект также предоставляет возможность совместного редактирования документов и хранения разных версий одного документа.

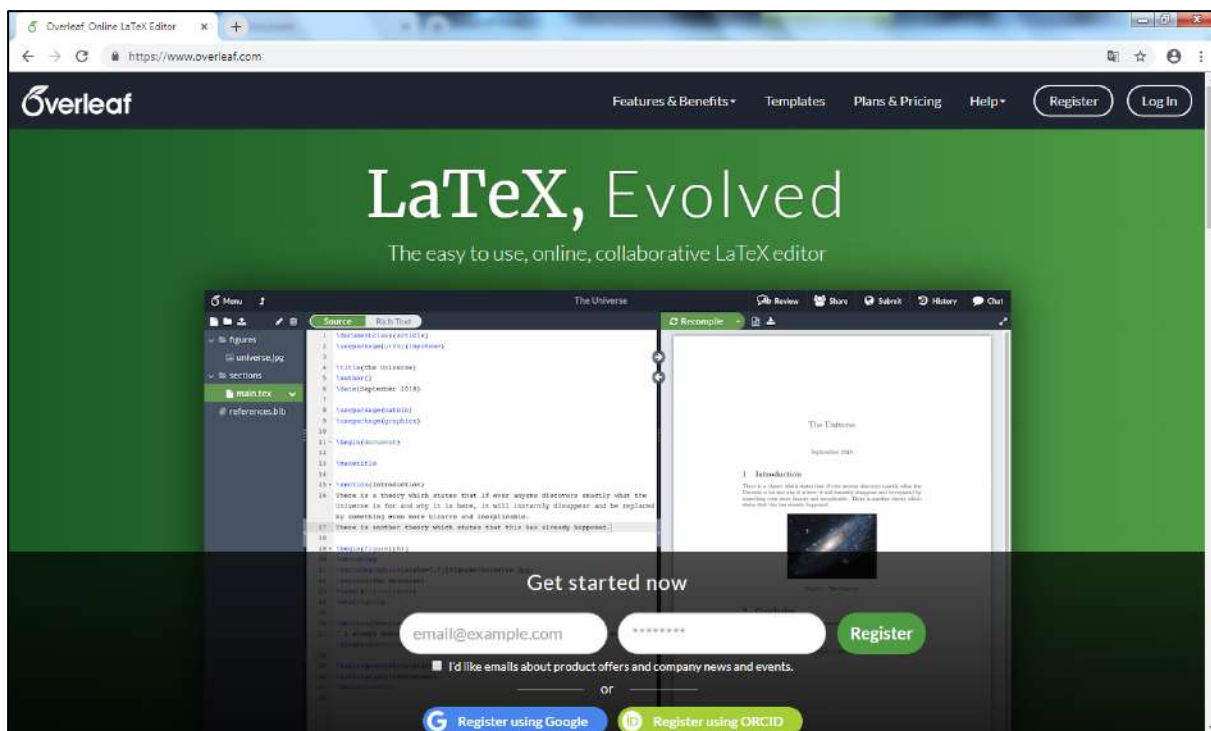


Рис. 1.40. Проект Overleaf.com.

Мы обуславливают выбор указанного ресурса следующими причинами:

1. Не требуется установка дистрибутива для работы с LaTeX. Весь процесс обработки и выдачи результата происходит удаленно, на серверах ресурса. Достаточно только подключения к сети Интернет.
2. Возможность совместного (удаленного) редактирования одного документа.
3. Быстро растущий проект с коллективом опытных разработчиков. При необходимости можно связаться с командой поддержки, или даже стать частью этой команды.
4. Большое число готовых шаблонов разных типов документов. Любой пользователь может загрузить собственный пример или шаблон, способствуя тем самым развитию проекта.
5. Регистрация для работы не требуется. Однако зарегистрированный пользователь получает ряд преимуществ, в частности – возможность хранить проекты и файлы к ним.
6. Использование совершенно бесплатно! (Доступны также и платные дополнительные услуги)

Вопросы для самопроверки

1. Почему LaTeX не позволяет просматривать конечный вид документа?
2. Перечислите основное программное обеспечение, необходимое для работы с LaTeX.
3. Обоснуйте важность использования текстовых редакторов с подсветкой синтаксиса команд.
4. Может ли дистрибутив содержать компоненты, делающие из него полноценное средство разработки документа? Если да, то приведите примеры.
5. Какие преимущества дает пользователю использование веб-сервисов по работе с LaTeX?

Практикум

1. Дистрибутив по LaTeX

1. Найдите описание возможностей TeX Live, MiKTeX и MacTeX. Какой из перечисленных дистрибутивов удовлетворяет системным требованиям Вашего ПК?
2. Изучите возможности редакторов TeXstudio, TeXworks, WinEdt, Sublime Text 3. В каких из них реализованы визуальные элементы?

2. Проект Overleaf.com

1. Перейдите на сайт Overleaf.com.
2. Какие возможности он предоставляет пользователям? Какие услуги доступны за дополнительную плату?

Глава 2

ОСНОВЫ LATEX

2.1 Структура документа

2.1.1 Hello, LaTeX!

По традиции, сложившейся у программистов, первая программа должна содержать наименьшее число команд, выводящих приветственное сообщение.

В нашем случае это будет не программа, а разметка страницы «статьи» с соответствующим текстом.

Код

```
\documentclass{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\begin{document}
    Привет, \LaTeX!
\end{document}
```

Указанный код сгенерирует лист с обозначенным текстом, при этом включается автоматическая нумерация страниц, задаются некоторые стандартные параметры листа.

Подробное описание команд и запуск проекта рассматривается далее.

2.1.2 Второй пример

Рассмотрим более сложный пример (результат см. на рис. 2.1).

Код

```
\documentclass{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}
\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}
\usepackage{graphicx}

\title{Введение в систему \LaTeX}
\author{Якубович Д.А.}
\date{\today}

\begin{document}
  \maketitle
  \section{\LaTeX\ для рядового пользователя}
  \LaTeX (<<латех>>) -- наиболее популярный набор
  макрорасширений системы компьютерной вёрстки
  \TeX, который облегчает набор сложных документов и
  Доступен широкому кругу пользователей.

  \section{\LaTeX\ для математиков и физиков}
  \LaTeX\ позволяет оформлять формулы с использованием
  специальных команд. Он является своего рода
  стандартом в типографии точных наук.

  Например, формула может быть указана в тексте:
   $f(x)=x^3+1$ . Также она может быть задана отдельной
  строкой и автоматически пронумерована:

  \begin{equation}\label{energy}
    \int\limits_{\sigma} d\omega =
    \int\limits_{\partial\sigma} \omega
  \end{equation}

  На формулу можно ссылаться, используя перекрестные
  ссылки: ссылка на формулу \eqref{energy}
\end{document}
```

Результат

Введение в систему ЛАТ_ΕΧ

Якубович Д.А.

12 мая 2019 г.

1 ЛАТ_ΕΧ для рядового пользователя

ЛАТ_ΕΧ («латех») – наиболее популярный набор макрорасширений системы компьютерной вёрстки Т_ΕΧ, который облегчает набор сложных документов и доступен широкому кругу пользователей.

2 ЛАТ_ΕΧ для математиков и физиков

ЛАТ_ΕΧ позволяет оформлять формулы с использованием специальных команд. Он является своего рода стандартом в типографии точных наук.

Например, формула может быть указана в тексте: $f(x) = x^3 + 1$. Также она может быть задана отдельной строкой и автоматически пронумерована:

$$\int_{\sigma} d\omega = \int_{\partial\sigma} \omega \tag{1}$$

На формулу можно ссылаться, используя перекрестные ссылки: ссылка на формулу (1)

Рис. 2.1. Окончательный вид документа из второго примера.

Для неподготовленного пользователя приведенный код выглядит весьма трудным. Однако на самом деле работать с ним несложно.

Из примера видно, что команды участвуют в разделении текста на разделы, абзацы; определенные группы команд оформляют математические формулы и символы. При этом некоторые операции по форматированию не прописаны вовсе: LaTeX сам оформляет документ по шаблону!

Текст разметки можно разделить на две основные части:

- **преамбула** документа: здесь размещается информация о типе документа, его структуре, оформлении, данные для ключевых разделов, а также подключаются классы («пакеты»);
- **тело** документа: содержит непосредственно текст документа, а также команды, определяющие его логическую структуру и форматирование.

Уже в приведенном примере документа можно обратить внимание на высокое качество полиграфии.

Вопросы для самопроверки

1. Что указывают в преамбуле документа?
2. Что понимают под «телом документа»?
3. Как LaTeX определяет настройки гарнитуры шрифта, оформление текста и документа в целом, если они не описаны в преамбуле?

Практикум

1. В Блокноте или Notepad++ создайте новый текстовый файл, скопируйте в него код разметки из первого примера. Сохраните его. При сохранении выберите формат .TEX.
2. По аналогии сохраните создайте файл для второго примера.
3. Измените в разметке второго примера автора на свои инициалы.

2.2 Работа с Overleaf.com

Для работы с LaTeX (в частности, с этим учебным курсом) вполне достаточно будет ограничиться работой с ресурсом Overleaf.com. Ресурс предоставляет возможность работы как без регистрации, так и с регистрацией аккаунта. Наличие последнего дает дополнительные возможности для пользователя, например, хранение файлов проекта.

2.2.1 Регистрация аккаунта

Шаг 1. Переходим на сайт [Overleaf.com](https://overleaf.com).

Шаг 2. На главной странице нажимаем на **Sign Up** («Зарегистрироваться»):

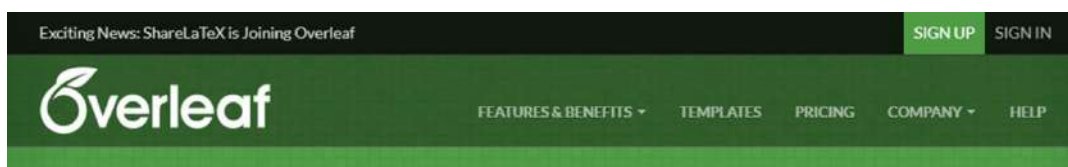


Рис. 2.2. Регистрация нового аккаунта.

Шаг 3. Вводим имя учетной записи (любое) и почтовый ящик, к которому будет привязан аккаунт. Далее нажимаем на **Create my account** («Создать мой аккаунт»):

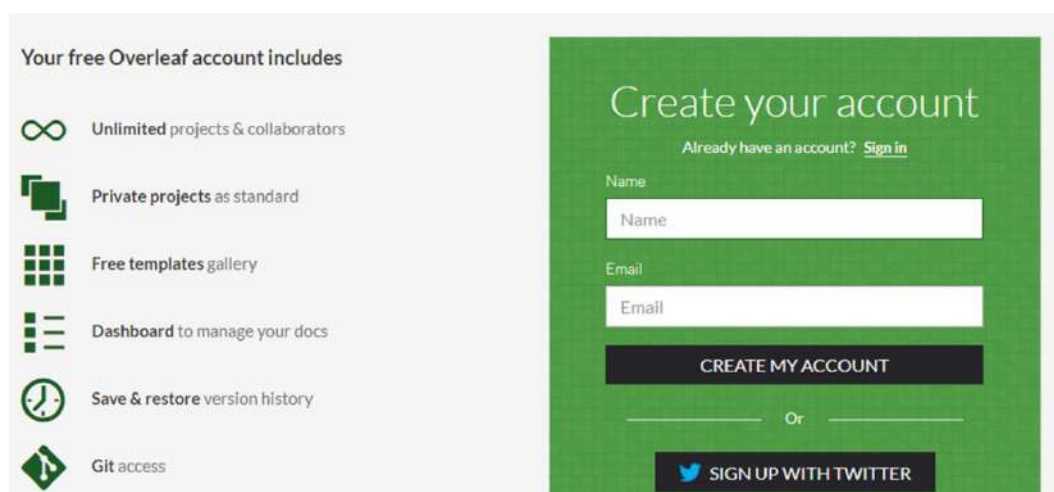


Рис. 2.3. Заполнение данных для нового аккаунта.

Шаг 4. Overleaf открывает страницу вашего аккаунта:

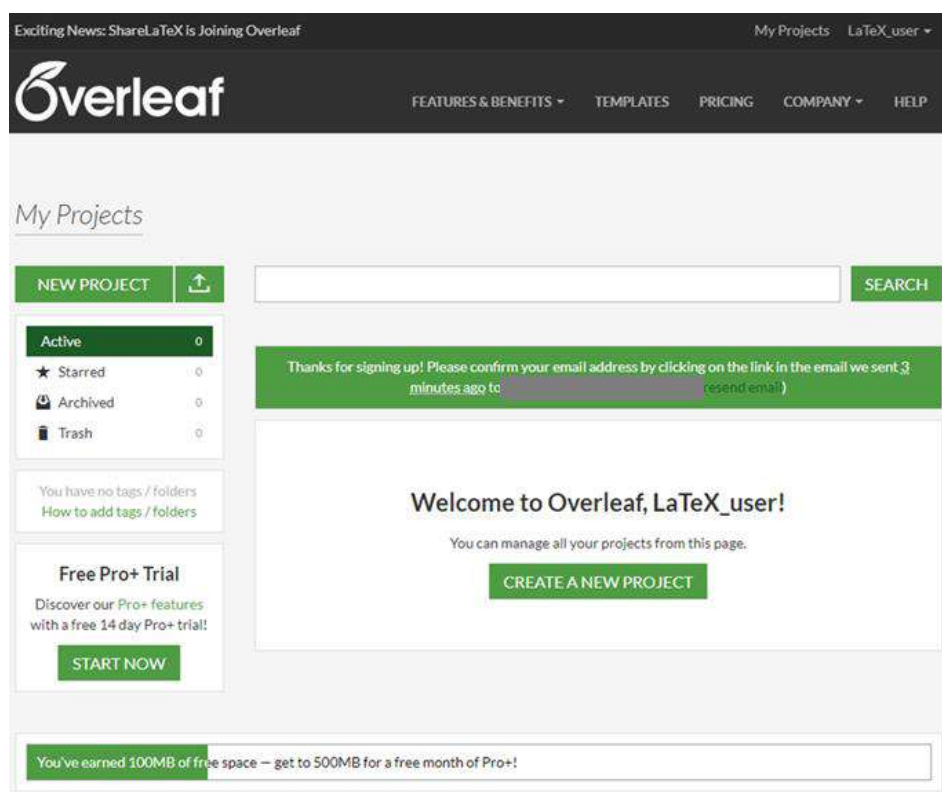


Рис. 2.4. Личный кабинет пользователя Overleaf.com.

Также на указанный почтовый ящик приходит письмо с просьбой подтверждения регистрации (по ссылкам в письме пока не следует переходить).

Шаг 5. Необходимо задать пароль. Для этого выйдите из аккаунта и перейдите по ссылке из письма (см. шаг 4):

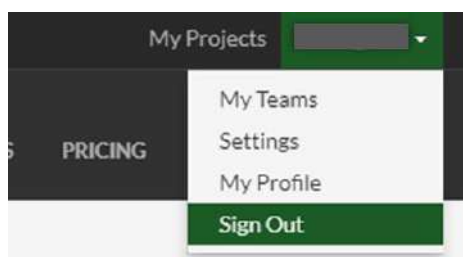


Рис. 2.5. Выход из системы.

В целях безопасности задайте пароль, отличный от вашего почтового!

Вводим придуманный
пароль

Повторяем пароль

Нажимаем на кнопку

Choose your password

To complete your registration, please choose a password.

Password

Confirm password

SET MY PASSWORD

Рис. 2.6. Создание пароля.

Шаг 6. После подтверждения пароля открывается окно личного кабинета. Также на почту приходит сообщение о смене пароля.

Шаг 7. Проверьте работу вашей учетной записи. Для этого выйдите из нее нажатием на кнопку **Sign Out**.

Далее выберите **Sign In** («Войти в систему») и задайте придуманные логин и пароль. Теперь ваш аккаунт полностью зарегистрирован.

2.2.2 Функции пользователя

Количество проектов (активных,
заархивированных и в корзине)

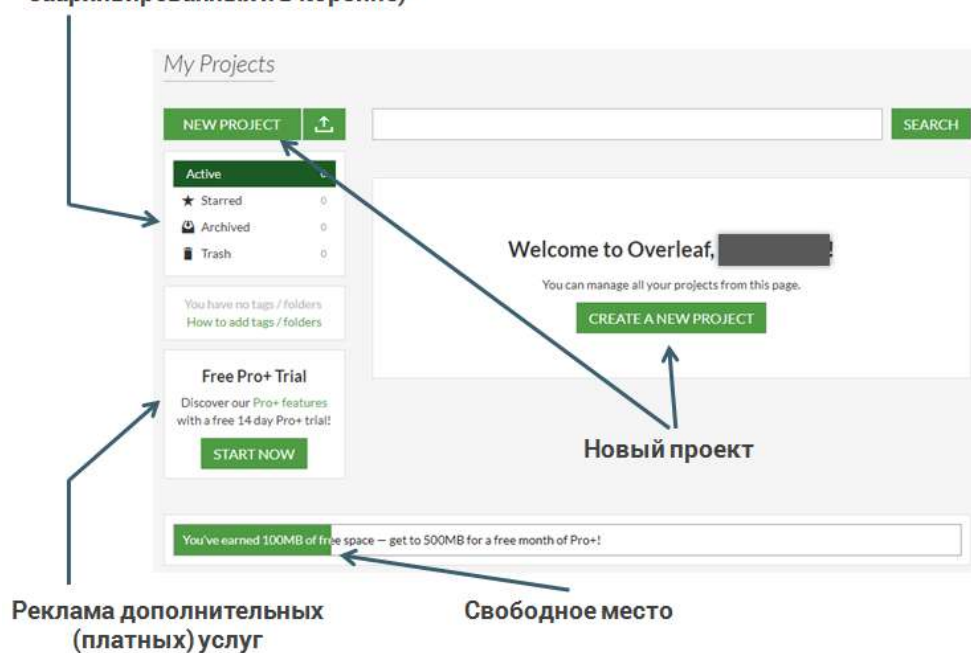


Рис. 2.7. Функции, доступные пользователю в личном кабинете.

При создании нового проекта можно выбрать один из шаблонов:

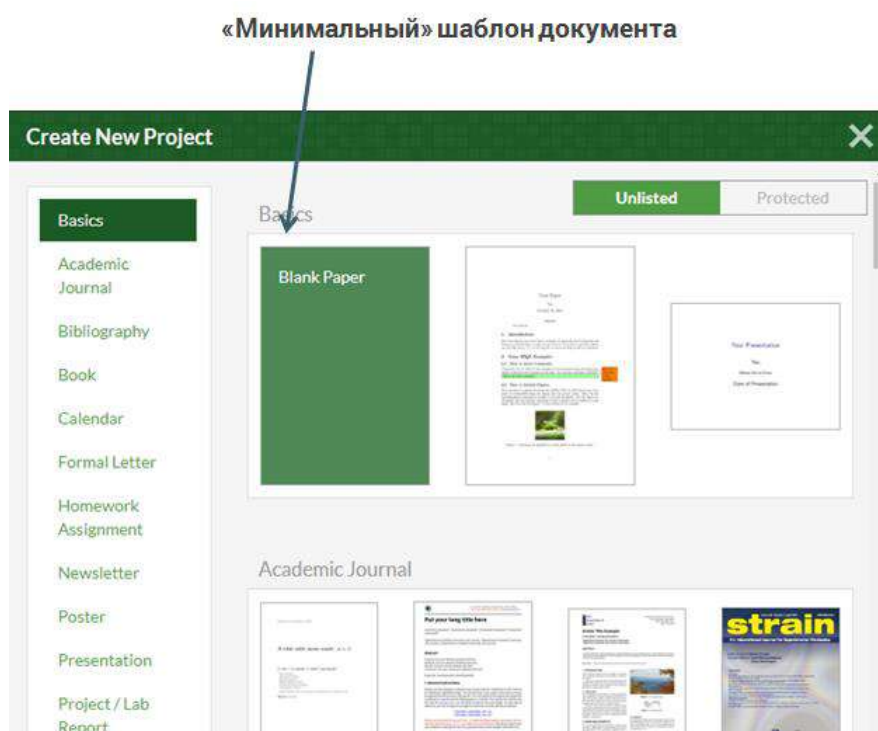


Рис. 2.8. Выбор шаблона кода и оформления нового документа.

Редактируемые проекты автоматически сохраняются в личном кабинете:

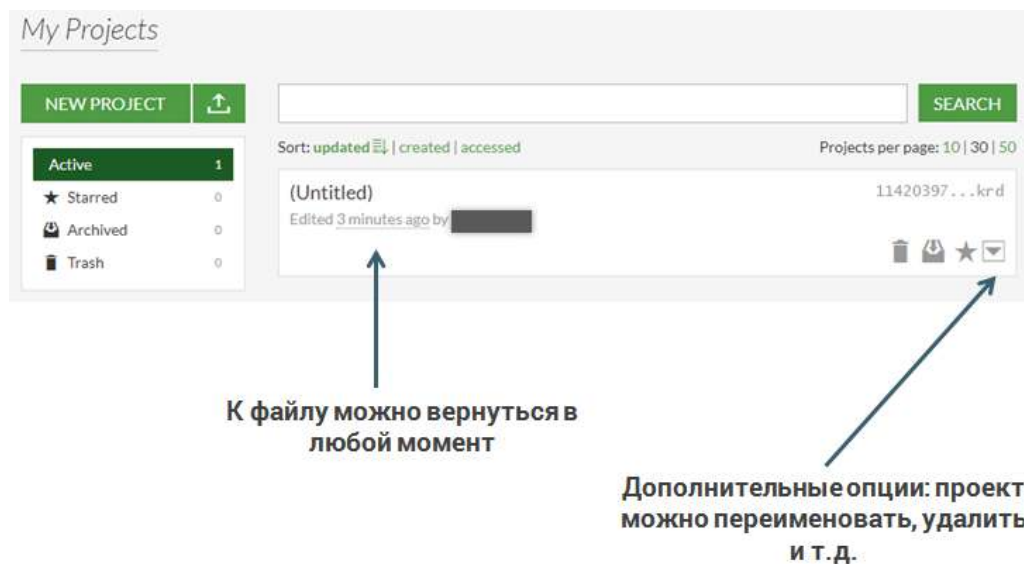


Рис. 2.9. Сохраненные проекты.

На вкладке Projects отображается структура проекта. Можно добавлять новые файлы, каталоги, в т.ч. и с ПК:

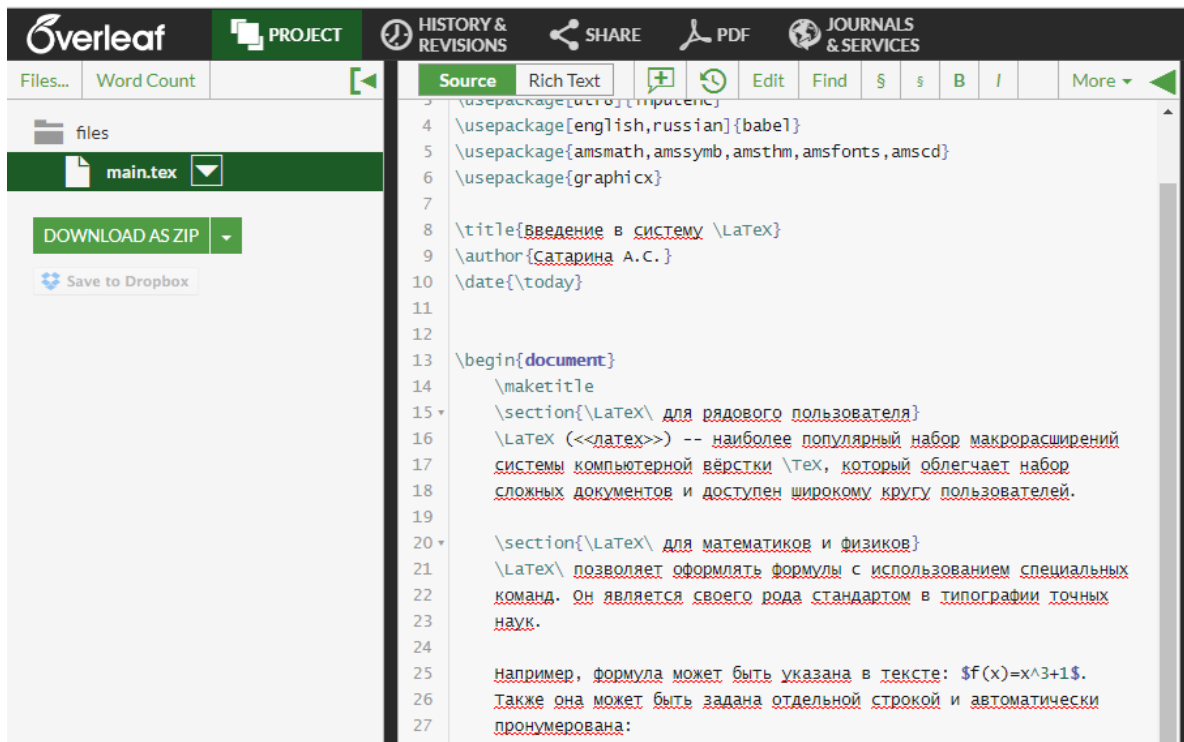


Рис. 2.10. Структура проекта и окно для редактирования кода разметки.

Рабочая среда состоит из двух областей:

1. левая содержит редактируемый текст (TEX-разметку);
2. правая отображает dvi-файл:

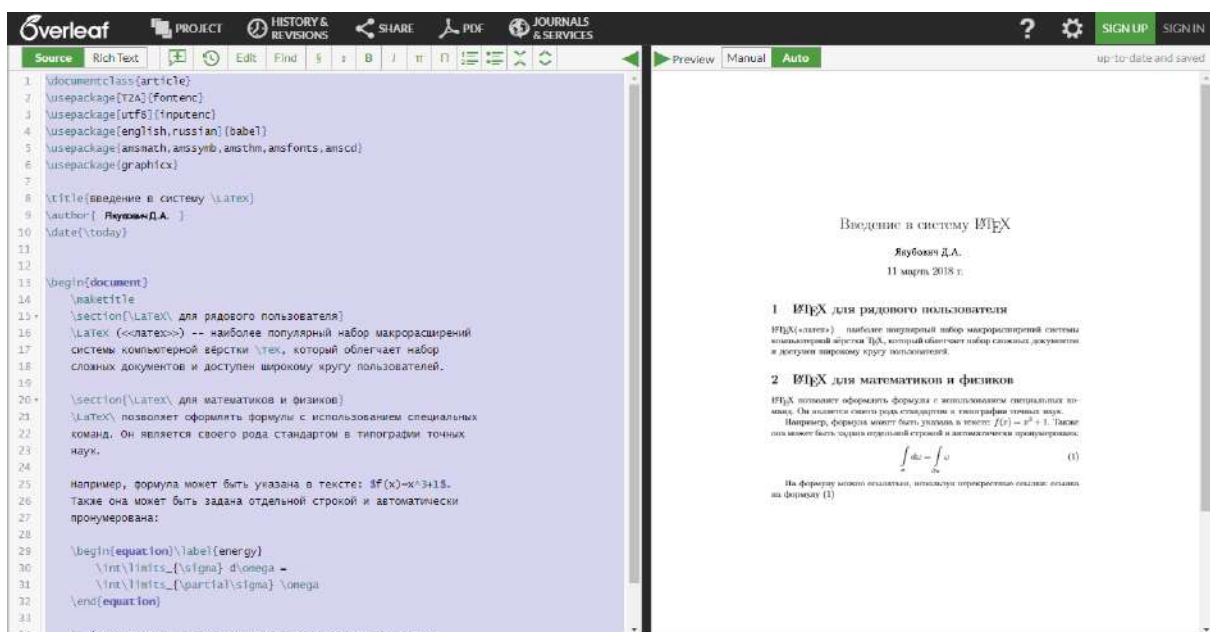


Рис. 2.11. Основной режим работы: слева – код разметки документа, справа – вид документа.

Обратите внимание, что редактор поддерживает два режима редактирования (правый верхний угол):

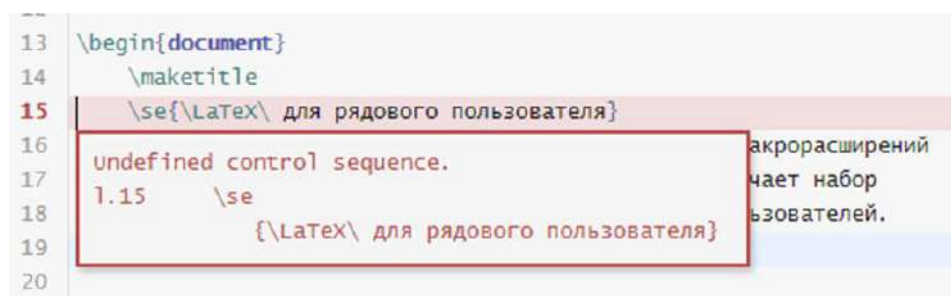
1. Rich Text («полувизуальный»);
2. Source (исходный код LaTeX).

Здесь и далее мы работаем с исходным кодом (первый режим мы рекомендуем использовать в том случае, когда читатель в достаточной мере овладеет техникой работы с системой)!

Для работы с нашим примером его код достаточно скопировать в левое окошко и дождаться результата в правом.

При щелчке на любом месте документа в правой части, через некоторое время редактор поместит курсор в ту строку кода, которая содержит соответствующий текст или команду разметки.

Преднамеренно допустите ошибку в коде, например, удалите несколько букв из команды `\section`:



```
13 \begin{document}
14   \maketitle
15   \se{\LaTeX\ для рядового пользователя}
16
17   undefined control sequence.
18   1.15   \se
19           {\LaTeX\ для рядового пользователя}
20
```

Рис. 2.12. Предупреждение об ошибках в коде разметки.

Компилятор LaTeX выдаст сообщение об ошибке и номер строки. В данном случае «Undefined control sequence» означает «Неопределенная управляющая последовательность», т.е. `\se` не является командой.

Замечание. В ряде ситуаций описание ошибок может быть не столь информативным.

Вы можете внести свои изменения в текст, например, удалить разметку одной из секций. Каждое изменение в редакторе фиксируется и отображаемое в правой части обновляется. Режим просмотра можно переключить из автоматического в ручной (ползунок в правом верхнем углу):



Рис. 2.13. Режимы компиляции и предварительного просмотра документа.

Наиболее часто возникающая проблема работы с Overleaf.com – длительный процесс компиляции DVI-файла (визуального вида). В этом случае:

- *скопируйте весь код в буфер клавиатуры или лучше в блокнот;*
- *обновите страницу (клавиша F5);*
- *вставьте код обратно и дождитесь результата.*

Overleaf поддерживает интеллектуальный режим работы редактора кода: он способен выдавать всплывающие подсказки и автоматически завершать написание команды (рис. 2.14). Для этого начните ввод команды, в выпадающем списке найдите требуемую. По нажатию клавиши **TAB** или **ПКМ** команда автоматически завершается.

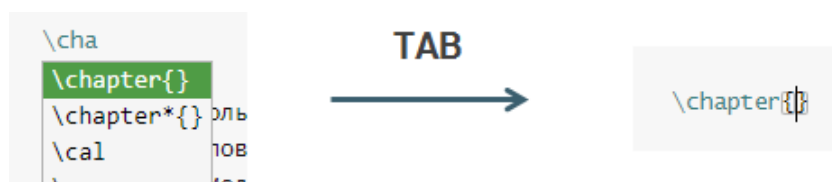


Рис. 2.14. Overleaf.com поддерживает режим интеллектуальной подсказки.

Вернитесь к исходному коду. На панели меню нажмите на кнопку **PDF**. Overleaf сгенерирует и сохранит на компьютер PDF версию документа.

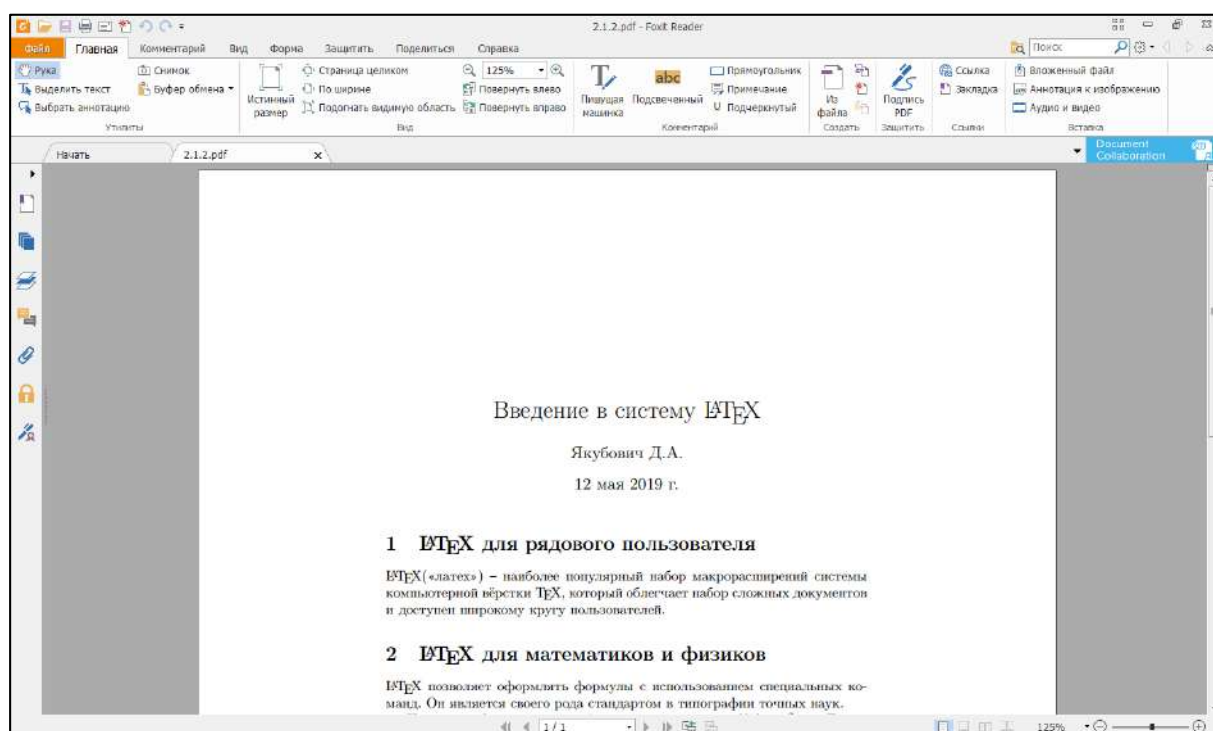



Рис. 2.15. Окончательная версия документа, экспортированная в PDF.

Для настройки среды нажмите на символ шестеренки:

- 1 Проверка орфографии. Можно включать и отключать. Орфография проверяется по указанному языку (слова других языков будут считаться некорректными и подчеркиваться красной волнистой линией).
- 2 Опции по сборке и просмотру (менять не требуется).
- 3 Настройка стиля редактора.
- 4 Сохранение изменений.

? 

Settings

Project Settings

(Apply to the current project only)

Project Title in My Projects

☒ Set Automatically from \title Command

Spell Check

☒ Enable

1 Spell Check Language

Русский язык (Россия)

Advanced Build Options

LaTeX Engine Choose Automatically

Preview Format Choose Automatically

SAVE PROJECT SETTINGS

4

My Settings

(Apply to all your projects)

Editor Mode (Key Map)

Default

Editor font size

Default A 3

☒ Auto-Close Brackets

☒ Auto-Complete Commands

Editor Theme

Default

```

1 % Here is some sample LaTeX:
2 \section{notation}
3 Let  $x_1, x_2, \dots, x_n$  be
4 \begin{equation}
5

```

SAVE MY SETTINGS

4

Рис. 2.16. Дополнительная настройка редактора.

Вопросы для самопроверки

1. Какие функции предоставляет пользователям ресурс Overleaf.com? Что доступно бесплатно?
2. Что делает Overleaf при выборе шаблона нового документа?
3. В каких форматах можно сохранить итоговый документ?

Практикум

1. Регистрация на сайте Overleaf.com

1. Согласно описанному алгоритму регистрации, создайте свой аккаунт на ресурсе Overleaf.com.
2. Проверьте работу своего аккаунта, изучите его возможности.

2. Запуск первого примера

1. Создайте пустой шаблон документа и скопируйте в него первый пример.
2. Сохраните документ в формате PDF.
3. Проверьте наличие проекта в Вашем аккаунте.

3. Доработка готового документа

1. Создайте второй проект и скопируйте код из второго примера.
2. Намеренно допустите в нем ошибку. Удастся ли Вам по описанию понять, где именно она допущена?
3. Дополните пример разметкой третьего пункта так, чтобы получить следующий результат:

Введение в систему \LaTeX

Якубович Д.А.

7 октября 2019 г.

1 \LaTeX для рядового пользователя

\LaTeX («латех») – наиболее популярный набор макрорасширений системы компьютерной вёрстки \TeX , который облегчает набор сложных документов и Доступен широкому кругу пользователей.

2 \LaTeX для математиков и физиков

\LaTeX позволяет оформлять формулы с использованием специальных команд. Он является своего рода стандартом в типографии точных наук.

Например, формула может быть указана в тексте: $f(x) = x^3 + 1$. Также она может быть задана отдельной строкой и автоматически пронумерована:

$$\int_{\sigma} d\omega = \int_{\partial\sigma} \omega \quad (1)$$

На формулу можно ссылаться, используя перекрестные ссылки: ссылка на формулу (1)

3 \LaTeX для учителей и преподавателей

Несмотря на широкую известность издательской системы \LaTeX , она крайне малопопулярна в среде работников сферы образования. В первую очередь это связано с более высоким «порогом вхождения» в систему.

На самом деле основы работы с \LaTeX может освоить даже школьник! Она во многом похожа на работу с HTML-кодом (версткой веб-страниц). Изначально это кажется неудобным, но со временем пользователь понимает всю мощь системы.

Чем же \LaTeX может быть полезен педагогу? Благодаря этой системе можно разрабатывать тексты лекций, пособия, книги, презентации и многое другое. Главное — все это система формирует в высоком полиграфическом качестве.

Рис. 2.17. Окончательный вид документа.

4. Знакомство с ShareLaTeX

1. Перейдите на сайт ShareLaTeX.com.
2. Зарегистрируйтесь на указанном ресурсе, выбрав бесплатный тариф. Для регистрации достаточно указать адрес электронной почты и придумать пароль (в целях безопасности придумайте пароль, отличный от почтового).
3. Создайте новый проект и реализуйте в нем исходное задание из пункта 2.1.2.

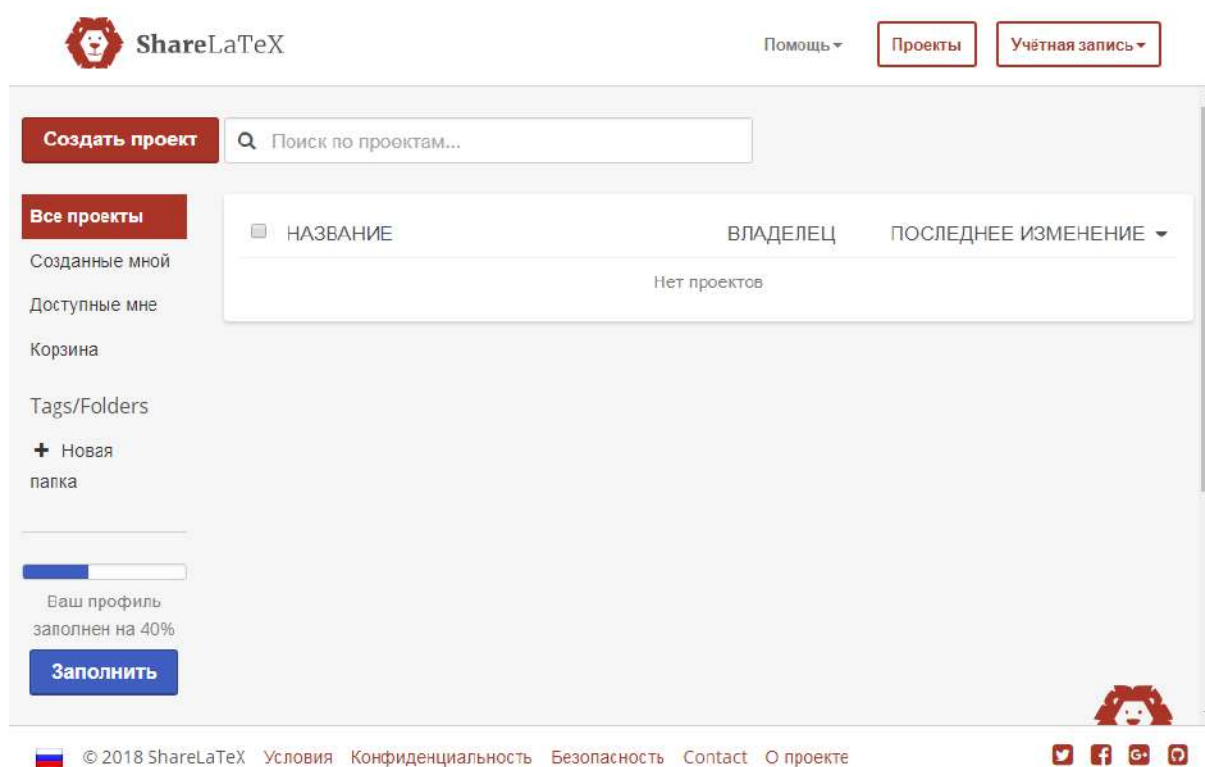


Рис. 2.18. Личный кабинет пользователя на сайте ShareLatex.com.

2.3 Обзор команд

2.3.1 Пример текста с математической символикой

Мы уже неоднократно отмечали преимущество LaTeX в вопросах верстки документов, содержащих математические символы и формулы. В этом разделе мы рассмотрим фрагмент текста по математическим методам в психологии. Таким образом подчеркивается, что LaTeX будет эффективен для набора любого текста.

Код

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\title{Математические методы в психологии}
\author{Якубович Д.А.}
\date{\today}

\linespread{1.15}

\begin{document}
  \maketitle

  \section{Числовые характеристики распределений}
  \subsection{Мода}
  \textbf{Мода} --- наиболее часто встречающееся
  числовое значение в выборке. Обозначается  $\hat{X}$ .

  \subsection{Медиана}
  \textbf{Медиана} --- значение, которое делит
  упорядоченное множество числовых данных пополам.
  Обозначается  $Md(X)$ .

  \subsection{Среднее арифметическое}
  \textbf{Среднее арифметическое} ряда из  $n$  числовых
  значений  $X_1$ ,  $X_2$ ,  $\ldots$ ,  $X_n$  вычисляется как

  \begin{equation}
    \label{AMean}
    \overline{X} = \frac{X_1 + X_2 + \cdots + X_n}{n} =
    \frac{1}{n} \sum_{k=1}^n X_k.
  \end{equation}

  Символ суммы в формуле ( $\ref{AMean}$ ) используется для
  сокращения записи.
\end{document}
```

Результат отображен на рис. 2.19.

Результат

Математические методы в психологии

Якубович Д.А.

11 марта 2018 г.

1 Числовые характеристики распределений

1.1 Мода

Мода — наиболее часто встречающееся числовое значение в выборке. Обозначается \hat{X} .

1.2 Медиана

Медиана — значение, которое делит упорядоченное множество числовых данных пополам. Обозначается $Md(X)$.

1.3 Среднее арифметическое

Среднее арифметическое ряда из n числовых значений X_1, X_2, \dots, X_n вычисляется как

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n} = \frac{1}{n} \sum_{k=1}^n X_k. \quad (1)$$

Символ суммы в формуле (1) используется для сокращения записи.

Рис. 2.19. Окончательный вид документа.

2.3.2 Общее описание документа

Приведенный выше код создает документ по стилю оформления научной статьи. В документе содержится информация о работе, авторе, он имеет четкую структуру в виде названия секций и подсекций, нумерация которых генерируется автоматически.

По умолчанию текст выравнивается по ширине, включен автоперенос русских и английских слов, страницы автоматически нумеруются.

2.3.3 Команды и символ \

Разметка LaTeX состоит из специальных команд, начинающихся с символа «\» и непосредственно **текста**.

Команды могут отличаться способом записи, назначением. Некоторые имеют **параметры** (обозначаются в фигурных скобках) и необязательные **опции** (в прямоугольных скобках).

LaTeX различает регистр букв! Например, `\large` и `\Large` являются различными командами.

2.3.4 Класс документа

Первая команда определяет **тип документа**. Каждый тип задает определенные стандарты форматирования, которые можно настраивать при необходимости.

```
\documentclass[12pt]{article}
```

Далее эта важная команда рассматривается подробно.

2.3.5 Подключение пакетов

Команда `\usepackage` подключает дополнительные пакеты LaTeX. Они расширяют стандартные возможности оформления или упрощают работу.

В данном примере подключаются пакеты **fontenc** и **inputenc**, отвечающие за кодировку символов кириллицы. Кодировка UTF-8 является наиболее современной и позволяет кодировать огромное количество символов.

Пакет **babel** организующий автоматические переносы слов; в опциях указывают один или более языков, для которых нужно делать переносы.

```
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}
```

Эти команды мы будем применять во всех примерах.

2.3.6 Титульный лист

Для формирования титульного листа могут использоваться специальные команды. Так, часть команд в преамбуле документа содержат информацию об авторе, названии документа и дате его последней редакции:

```
\title{Математические методы в психологии}
\author{Якубович Д.А.}
\date{\today}
```

. . .

```
\maketitle
```

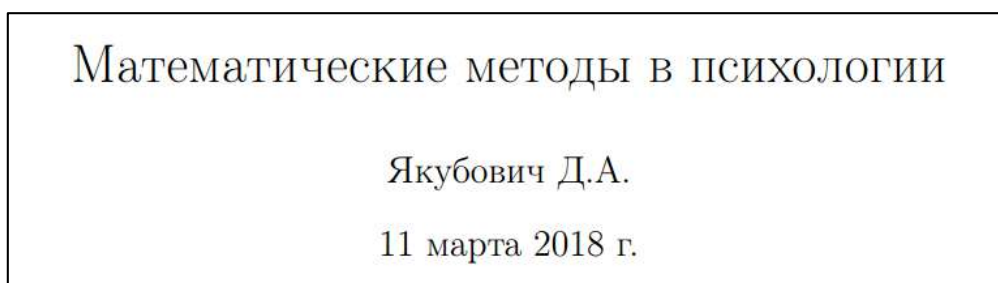


Рис. 2.20. Оформление заголовка документа, автора и даты.

Команда внутри тела документа **\maketitle** компоует данные этих команд и отображает их в документе согласно заданному в нача-

ле стилю. Для книг или докладов титульный лист формируется отдельной страницей. Впрочем, эти команды не являются обязательными: заголовок может быть оформлен совершенно по-другому.

2.3.7 Комментарии

В коде разметки документа можно оставлять пояснения, которые игнорируются компилятором. Их называют комментариями.

Комментарий начинается с символа «%». Все, что находится справа от этого символа, считается комментарием и не отображается в итоговом документе.

```
% Данные для титула
\title{Математические методы в психологии}
\author{Якубович Д.А.}
\date{\today}

\linespread{1.15}    % междустрочный интервал
```

Важно отметить, что комментарий в LaTeX действует только в рамках одной строки.

2.3.8 Рубрикация документа

Документ LaTeX имеет четкое разделение на части или главы, секции, параграфы. Для этого существуют специальные команды рубрикации.

```
\section{Числовые характеристики распределений}
\subsection{Мода}
```



Рис. 2.21. Автоматическая рубрикация.

В каждом типе документов предусмотрена своя иерархия. Например, для статей нет частей и глав.

Секционирование документа автоматически размечает заголовки и задает им нумерацию. На базе секций можно построить оглавление и оформить колонтитулы.

2.3.9 Текст и математические символы

Текст может быть оформлен с помощью специальных команд и деклараций: от гарнитуры шрифта до положения и элементов декора.

Особое место в LaTeX занимают математические символы. Для них используется специальное оформление и шрифты (см. рис. 2.22).

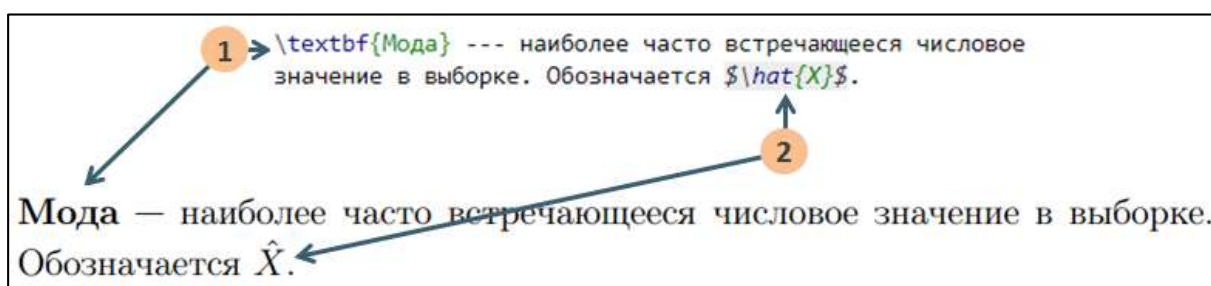


Рис. 2.22. Специальный синтаксис для оформления формул и математических символов в LaTeX.

Работа с формулами заслуживает отдельного внимания (мы вернемся к этому вопросу, когда рассмотрим наиболее важные элементы и команды LaTeX).

2.3.10 Окружения

Окружение — это команда, осуществляющая работу над элементами внутри следующих команд:

```
\begin{имя_окружения}
```

```
\end{имя_окружения}
```

Указанное в скобках окружение дает информацию о том, как нужно форматировать текст внутри.

Так, в примере используется окружение **equation**, позволяющее оформлять выделенные отдельным абзацем формулы. В отличие от обычной формулы, это окружение автоматически добавляет номер к формуле.

2.3.11 Перекрестные ссылки

Перекрестные ссылки в LaTeX могут быть организованы с различными элементами документа. Преимущество перед другими офисными пакетами – автоматическое обновления ссылки при изменении положения источника в тексте.

Указатель ссылки создается с помощью команды `\label`. Команда `\ref` подставляет указатель ссылки в требуемое место документа.

Существуют разные типы ссылок. В частности, допускается создание интерактивных ссылок. В нашем примере используется ссылка на формулу:

```
\begin{equation}
  \label{AMean}
  \overline{X} = \frac{X_1 + X_2 + \cdots + X_n}{n} =
  \frac{1}{n} \sum\limits_{k=1}^n X_k.
\end{equation}
```

Символ суммы в формуле (`\ref{AMean}`) используется для сокращения записи.

1 → `\begin{equation}`
`\label{AMean}`
`\overline{X} = \frac{X_1 + X_2 + \cdots + X_n}{n} =`
`\frac{1}{n} \sum\limits_{k=1}^n X_k.`
`\end{equation}`

Символ суммы в формуле (`\ref{AMean}`) используется для сокращения записи.

2 ↑

$$\overline{X} = \frac{X_1 + X_2 + \cdots + X_n}{n} = \frac{1}{n} \sum_{k=1}^n X_k. \quad (1) \leftarrow 1$$

Символ суммы в формуле (1) используется для сокращения записи.

2 ↑

Рис. 2.23. Перекрестные ссылки и автоматическая нумерация формул.

Вопросы для самопроверки

1. Какие особенности синтаксиса имеют команды LaTeX?
2. Для чего подключаются пакеты?
3. Приведите примеры ситуаций, когда использование комментариев в разметке может быть важным.
4. Какие возможности поддерживают команды рубрикации?
5. Существует ли возможность работы с перекрестными ссылками и как она может быть организована?
6. Почему рекомендуется придерживаться определенного стиля оформления кода разметки документа?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Доработка файла

1. Согласно примеру предыдущего занятия, создайте новый документ с соответствующей разметкой.
2. Укажите свои ФИО, дату измените вручную на текущую.
3. Дополните описание документа пунктом 4 (см. рис. 2.24-рис. 2.25).

2. Расширение файла

Потренируйтесь и добавьте в файл еще часть текста (произвольно). Для простоты возьмите обычный текст, без содержания математических символов, таблиц или изображений.

Математические методы в психологии

Якубович Д.А.

7 октября 2019 г.

1 Числовые характеристики распределений

1.1 Мода

Мода — наиболее часто встречающееся числовое значение в выборке. Обозначается \hat{X} .

1.2 Медиана

Медиана — значение, которое делит упорядоченное множество числовых данных пополам. Обозначается $Md(X)$.

1.3 Среднее арифметическое

Среднее арифметическое ряда из n числовых значений X_1, X_2, \dots, X_n вычисляется как

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n} = \frac{1}{n} \sum_{k=1}^n X_k. \quad (1)$$

Символ суммы в формуле (1) используется для сокращения записи.

Рис. 2.24. Окончательный вид документа (стр. 1).

1.4 Амплитуда ряда

Амплитудой числового ряда называют разницу между наибольшим и наименьшим значением: $A(x) = \max X - \min X$.

Например, для числового ряда $X = \{5, 0, -3, 1, 5, 2\}$ амплитуда равна $A(X) = 5 - (-3) = 8$.

Рис. 2.25. Окончательный вид документа (стр. 2).

2.4 Настройка документа

2.4.1 Классификация команд LaTeX

Разметка тела документа LaTeX определяется самим текстом и командами по его форматированию. Рассмотрим виды команд подробнее. Отметим, что предложенное нами подразделение команд весьма условно. Оно лишь приводится в целях упрощения понимания базового синтаксиса LaTeX.

В этом разделе в примерах для краткости не приводится весь код разметки документа, а только его тело. Для проверки примеров читатель может использовать любой из шаблонов, рассмотренных ранее.

Специальные символы

Все символы LaTeX делятся на две категории:

1. символы, которые отображаются так же, как и набраны (таких большинство, например буквы, цифры);
2. символы особого назначения; всего их 10:

{ } \$ & # % _ ^ ~ \

В большинстве своем эти символы являются частью некоторой команды. Чтобы их отобразить, как символы, необходимо приписать символ «\», например:

\% \{ \\$

(в этом случае говорят, что символ «экранируют»).

Пример разметки LaTeX

Специальные символы лучше не писать сами по себе, поскольку обычно это является ошибкой. Если вы хотите их отобразить, то припишите к ним символ обратного следа, например, `\$`, `\%` и т.д.

Результат

Специальные символы лучше не писать сами по себе, поскольку обычно это является ошибкой. Если вы хотите их отобразить, то припишите к ним символ обратного слеша, например, \$, % и т.д.

Рис. 2.26. Оформление специальных символов.

Команды

Команды LaTeX начинаются с обратного слеша «\». Есть **оди-
ночные команды**, а есть команды с **параметрами** и/или **опциями**.

Одиночные команды делятся на два типа:

1. Команды, состоящие из знака «\» и одного символа, не яв-
ляющегося буквой, например

`\$, \{, \#.`

2. Команды, состоящие из символа «\» и последовательности
букв, в т.ч. из одной, например

`\LaTeX, \TeX.`

Важно отметить, что даже при наличии пробела в правой части этих команд они пишутся слитно со следующим сло-
вом. Чтобы задать пробел, поставьте в конце команды сим-
вол «\»:

`\LaTeX\`

Пример разметки LaTeX

Язык разметки `\LaTeX\` может использовать даже школьник!
Опытных пользователей в шутку называют `\TeX` никами.

Результат

Язык разметки `LaTeX` может использовать даже школьник! Опытных
пользователей в шутку называют `TeX`никами.

Рис. 2.27. Команды в разметке.

Комментарии

Комментарий – это часть разметки документа, которая игнори-
руется компилятором и не отображается в документе. Комментарий

начинается с символа «%». Все, что находится справа от этого символа, считается комментарием.

Комментарии используются для пояснения фрагментов кода разметки, используемых команд, пометок для других верстальщиков (если документ оформляет несколько пользователей или редакторов). Комментарии также удобно использовать для «заморозки» команд.

Важно отметить, что комментарий в LaTeX действует только в рамках одной строки.

Пример разметки LaTeX

```
% Редактор: Якубович Д.А..  
% Дата последней редакции: 12.03.2019  
Комментарии принято писать в отдельной строке.  
Если поставить \% в середине предложения,  
% то этот текст вы не увидите.  
Часто комментарий пишут в конце команды, чтобы пояснить  
ее назначение. Также комментарий позволяет заморозить ко-  
манду, например:  
\section{Этот заголовок не печатается}
```

Результат

Комментарии принято писать в отдельной строке. Если поставить % в середине предложения, Часто комментарий пишут в конце команды, чтобы пояснить ее назначение. Также комментарий позволяет заморозить команду, например:

Рис. 2.28. Комментарии в разметке документа не отображаются в итоговом документе.

Команды с параметрами и опциями

Существуют команды, которые имеют параметры. В общем виде они описываются следующим образом:

```
\команда{параметр}
```

Обычно параметр – это объект, над которым совершается действие, либо дополнительная информация для команды.

Кроме параметров, команда может иметь и необязательные опции, обозначаемые в квадратных скобках:

```
\команда[опции]{параметр}
```

Опции указывают, как осуществить действие над параметром.

Команда может содержать несколько параметров или опций: в этом случае их перечисляют через запятую.

Пример разметки LaTeX

```
% Команда с одним параметром: задает секцию (заголовок)
\section{Введение}
```

```
% Команда с параметром и опцией: кодировка документа
\usepackage[utf8]{inputenc}
```

```
% Команда с двумя опциями и параметром:
% вкл. автоматические переносы слов
\usepackage[english,russian]{babel}
```

```
% Команда с четырьмя опциями и параметром.
% Можно писать в несколько строк
% Задает геометрию страницы
\usepackage[
    top=1cm,
    bottom=2cm,
    left=3cm,
    right=1.5cm
]{geometry}
```

```
% То же самое
\usepackage[top=1cm, bottom=2cm, left=3cm,
right=1.5cm]{geometry}
```

Группы

Группа – это часть текста разметки, заключенная в фигурные скобки.

```
{ группа }
```

Обычно группа ограничивает действие команды (например, форматирования текста), которая в ней находится. Однако некоторые команды, называемые глобальными, сохраняют свое действие и за пределами той группы, где они были употреблены.

Группы могут быть вложены друг в друга, что позволяет осуществлять несколько преобразований над фрагментом текста.

При работе с группой нужно аккуратно следить за ее скобками: незакрытая скобка может привести к ошибке компиляции или оформления.

Пример разметки LaTeX

Группа позволяет форматировать требуемую часть текста, например `{\bf}` сделать его жирным начертанием.} Если же не применять скобки группы, `\it` то команда форматирования действует либо до конца, либо до тех пор, `\rm` пока не встретится команда того же класса.

Результат

Группа позволяет форматировать требуемую часть текста, например **сделать его жирным начертанием.** Если же не применять скобки группы, *то команда форматирования действует либо до конца, либо до тех пор,* пока не встретится команда того же класса.

Рис. 2.29. Использование команды группировки.

Окружения

Окружение – это команда, осуществляющая работу над элементами внутри этой команды. Окружение можно понимать как ящик-преобразователь; оно имеет синтаксис:

```
\begin{имя_окружения}  
    % Содержимое  
\end{имя_окружения}
```

Окружения похожи на группы, т.е. они выполняют роль скобок. А указанное в скобках окружение дает информацию о том, как нужно форматировать текст или содержимое внутри.

Пример разметки LaTeX

```
\begin{center}  
    Выравнивает текст по центру.  
\end{center}  
  
\begin{flushright}  
    Выравнивает текст по правому краю.  
\end{flushright}
```

Окружение для формулы с автонумерацией:
`\begin{equation}`

```

c^2 = a^2 + b^2.
\end{equation}

\begin{equation}
\sin^2 x + \cos^2 x = 1.
\end{equation}

```

Результат

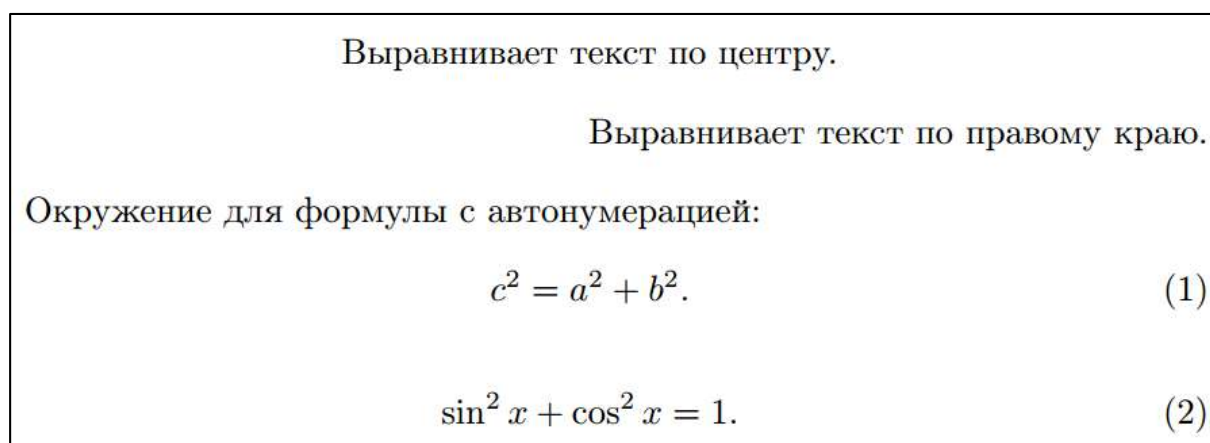


Рис. 2.30. Использование окружений для оформления текстовых абзацев и формул.

Параметры

LaTeX построен на базе TeX, поэтому пользователю доступен еще один класс команд, называемых **параметрами**. В качестве параметров выступают, например, величина абзацного отступа, ширина и высота страницы, расстояние по вертикали между соседними абзацами и т.д.

Все параметры имеют значение, предопределенное выбранным классом документа. Однако пользователь может поменять их стандартные значения по своему усмотрению следующим образом:

```
\параметр = величина
```

При этом:

- если параметр определен в преамбуле, то он действует во всем документе;
- если параметр определен в группе { }, то он действует только внутри группы.

Пример разметки LaTeX

```
% Глобальный отступ красной строки (описан в преамбуле)
```

```

\parindent = 1.25cm

\begin{document}
  Абзац №1 - отступ красной строки равен 1.25 см.

  {
    \parindent = 1cm    % Локальный отступ (в группе
    { })

    Абзац №2 - отступ красной строки равен 1 см.

    Абзац №3 - отступ красной строки равен 1 см.
  }

  Абзац №4 - отступ красной строки равен 1.25 см.
\end{document}

```

Результат

Абзац №1 - отступ красной строки равен 1.25 см.
 Абзац №2 - отступ красной строки равен 1 см.
 Абзац №3 - отступ красной строки равен 1 см.
 Абзац №4 - отступ красной строки равен 1.25 см.

Рис. 2.31. Изменение стандартных параметров оформления документа.

2.4.2 Класс документа

Первой и обязательной командой является обозначение класса документа.

Класс определяет шаблон оформления документа.

```
\documentclass[опции]{класс}
```

Название класса является обязательным параметром.

В зависимости от выбранного класса также неявно устанавливаются и некоторые опции по умолчанию (их можно менять, указывая явно). Класс определяет общую типографику страницы (например,

для книги есть колонтитул, для статьи по умолчанию он не генерируется). Это, в частности, влияет на количество страниц.

Классы могут быть стандартными, а также добавлены пользователем.

Примеры стандартных классов

- **article** – стиль научной статьи; не содержит разделения на главы; титульный лист с помощью команды `\maketitle` размещается вверху страницы;
- **report** – стиль для дипломных работ, диссертаций; делит документ на главы, титульный лист и оглавление печатается отдельными листами;
- **book** – стиль для оформления книг; по умолчанию компоновка (и печать) осуществляется с двух сторон листа;
- **letter** – деловое письмо;
- **slides** – стандартная презентация;
- **beamer** – современный пакет для верстки презентаций.

Примеры опций

Опции позволяют изменить ряд стандартных значений класса на требуемые.

- **10pt, 11pt, 12pt** – глобальный размер шрифта (по умолчанию 10pt);
- **a4paper, a5paper, b5paper, letterpaper** – размер бумаги (по умолчанию letterpaper);
- **landscape, portrait** – задает альбомную или портретную ориентацию бумаги (по умолчанию – portrait);
- **twocolumn** – вывод текста в две колонки;
- **fleqn** – формулы выравниваются влево, а не по центру;
- **leqno** – формулы нумеруются слева;
- **titlepage / notitlepage** – новая страница после заголовка (вкл./выкл.);
- **twoside, oneside** – определяет двустороннюю или одностороннюю печать;
- **openright, openany** – разрешает начинать главы только на нечетных страницах или на любых.

В качестве примера возьмем уже созданный ранее документ и посмотрим, как меняется его форматирование в зависимости от выбранного класса.

Пример 1

```
% Задаем стиль форматирования книги
% Опции по умолчанию: кегель - 10pt, размер - a4paper,
% колонки - onecolumn и др.)
\documentclass{book}
```

Пример 2

```
% Задаем стиль форматирования статьи
% Опции: кегель - 12pt, размер - a4paper,
% колонки - twocolumn (две)
\documentclass[a4paper, 12pt, twocolumn]{article}
```

2.4.3 Подключение дополнительных пакетов

Назначение и подключение пакетов

Синтаксис LaTeX допускает группировку команд и фактически создание новых на основе базовых. Это роднит его с языками программирования. Новая команда позволяет:

- сократить объем кода и повысить его читабельность;
- ускорить верстку;
- не тратить время на работу с тем, что уже реализовано.

Связанные общей задачей команды обычно группируются в отдельные пакеты, которые пользователь может подключать и использовать в своей работе.

Многие пакеты уже входят в состав современного дистрибутива LaTeX, некоторые поставляются отдельно (например, научными журналами, институтами, издательствами, компаниями и т.д.). Так или иначе – пользователь сталкивается с необходимостью их подключения.

```
\usepackage[опции]{пакет}
```

Подключает пакет. Можно подключить несколько пакетов, указав их через запятую.

Примеры пакетов

- **babel** – пакет языковой поддержки, в т.ч. руссификации;
- **inputenc** – пакет задания кодировки текста документа;
- **geometry** – пакет для настройки размеров страницы и ее полей;
- **graphicx** – пакет для импорта в документ изображений в различных форматах;
- **color, xcolor** – пакеты для стандартной и расширенной работы с цветами;
- **amsmath, amssymb, amsthm, amsfonts**, др. – пакеты для оформления математических статей по стандартам Американского математического общества (AMS);
- **hyperref** – пакет для создания гиперссылок в итоговом документе.

Примеры

% 1. Группа пакетов и опций для работы с кириллицей.

```
\usepackage[T2A]{fontenc}  
\usepackage[utf8]{inputenc}
```

% 2. Поддержка переноса слов для указанных языков

```
\usepackage[english,russian]{babel}
```

% 3. Стандарты оформления текста и элементов AMS

```
\usepackage{amsmath, amssymb, amsthm, amsfonts, amscd}
```

% 4. Подключение расширенных цветовых настроек и графики

```
\usepackage{xcolors}  
\usepackage{graphicx}
```

При работе с командой `\usepackage` следует придерживаться следующих правил:

1. Старайтесь подключать в первую очередь важнейшие пакеты.
2. Для удобства «родственные» пакеты подключайте одной командой (как в п. 3), а разные – последовательным перечислением (как в п. 1 и 4).
3. Пишите комментарии к подключаемым пакетам. Так вы всегда сможете вспомнить об их назначении.

2.4.4 Колонтитулы

`\pagestyle{стиль}`

Задаёт стиль колонтитулов для всего документа.

Возможные значения стиля:

- **plain** – номера страниц снизу по середине, колонтитулов нет (по умолчанию устанавливается для класса article);
- **headings** – название главы и номера страниц сверху (по умолчанию ставится классу book);
- **empty** – без колонтитулов и нумерации страниц.

`\thispagestyle{стиль}`

Задаёт стиль колонтитулов для текущей страницы.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Печатаем колонтитул (со второй страницы)
\pagestyle{headings}

\title{Математические методы в психологии}
\author{Якубович Д.А.}
\date{\today}

\linespread{1.15}

\begin{document}
    % текст разметки документа
    % (для краткости не приводим)
\end{document}
```

Результат

Математические методы в психологии

Якубович Д.А.

13 марта 2018 г.

1 Числовые характеристики распределений

1.1 Мода

Мода — наиболее часто встречающееся числовое значение в выборке. Обозначается \hat{X} .

Например, для ряда (1, 7, 0, 3, 7, 4, 7) модой является число 7.

1.2 Медиана

Медиана — значение, которое делит упорядоченное множество числовых данных пополам. Обозначается $Md(X)$.

Например, для выборки (1, 3, 4, 10, 15) медиана равна 4, а для выборки (2, 6, 7, 9, 12, 16) равна 8 (как среднее арифметическое двух «центральных» элементов).

1.3 Среднее арифметическое

Среднее арифметическое ряда из n числовых значений X_1, X_2, \dots, X_n вычисляется как

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n} = \frac{1}{n} \sum_{k=1}^n X_k. \quad (1)$$

Символ суммы в формуле (1) используется для сокращения записи.

Рис. 2.32. Использование стиля оформления документа с колонтитулом (стр. 1).

1.4 Разброс выборки

Разброс (размах) выборки обозначается символом R — это разность между максимальной и минимальной величиной вариационного ряда.

Чем сильнее варьируется значение измеряемого признака, тем больше значение R , и наоборот.

Рис. 2.33. Использование стиля оформления документа с колонтитулом (стр. 2).

2.4.5 Геометрия страницы

Для настройки размеров листа и его полей удобно использовать пакет **geometry**. Рассмотрим наиболее востребованные параметры этого пакета.

```
\usepackage[опции]{geometry}  
\geometry{сторона=отступ}
```

Следующие три примера демонстрируют абсолютно одинаковую настройку полей листа.

Пример разметки LaTeX

% Способ 1

```
\usepackage{geometry}  
\geometry{left = 3cm}  
\geometry{right = 1.5cm}  
\geometry{top = 2cm}  
\geometry{bottom = 3cm}
```

% Способ 2

```
\usepackage[left = 3cm, right = 3cm, top = 2cm, bottom =  
3cm]{geometry}
```

% Способ 3

```
\usepackage{geometry}  
\geometry{a4paper, total = {165mm,247mm}, left = 3cm, top  
= 2cm}
```

Следующая схема объясняет, почему все три способа дают одинаковый результат.

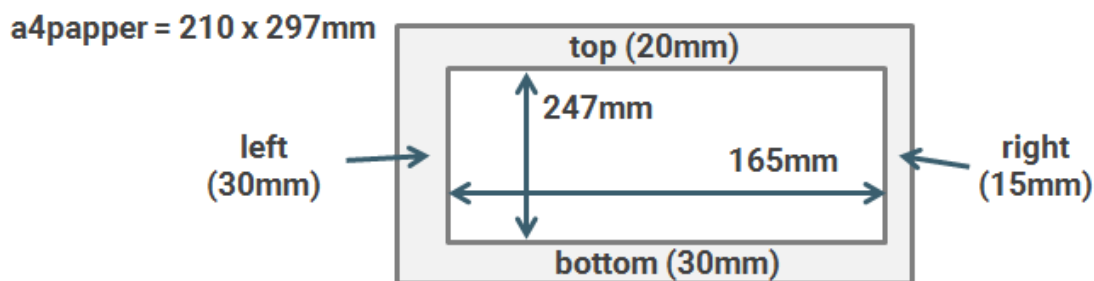


Рис. 2.34. Параметры настройки геометрии страницы.

Рассмотрим пример настройки параметров полей листа. Установим отступы по 3 см слева, справа и снизу и 2 см – сверху. Кроме того, поменяем ориентацию страницы на альбомную.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% настраиваем параметры страницы
\usepackage[
    landscape,          % альбомная ориентация листа
    left = 3cm,
    right = 3cm,
    top = 2cm,
    bottom = 3cm
]{geometry}

% информация об авторе
\title{Математические методы в психологии}
\author{Якубович Д.А.}
\date{\today}

% междустрочный интервал
\linespread{1.15}

\begin{document}

    % код разметки
    % (см. предыдущий пример)

\end{document}
```

Результат

Математические методы в психологии

Якубович Д.А.

14 марта 2018 г.

1 Числовые характеристики распределений

1.1 Мода

Мода — наиболее часто встречающееся числовое значение в выборке. Обозначается \hat{X} .

Например, для ряда (1, 7, 0, 3, 7, 4, 7) модой является число 7.

1.2 Медиана

Медиана — значение, которое делит упорядоченное множество числовых данных пополам. Обозначается $Med(X)$.

Например, для выборки (1, 3, 4, 10, 15) медиана равна 4, а для выборки (2, 6, 7, 9, 12, 16) равна 8 (как среднее арифметическое двух «центральных» элементов).

1.3 Среднее арифметическое

Среднее арифметическое ряда из n числовых значений X_1, X_2, \dots, X_n вычисляется как

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n} = \frac{1}{n} \sum_{k=1}^n X_k. \quad (1)$$

Символ суммы в формуле (1) используется для сокращения записи.

1

1.4 Разброс выборки

Разброс (размах) выборки обозначается символом R — это разность между максимальной и минимальной величиной вариационного ряда.

Чем сильнее варьируется значение измеряемого признака, тем больше значение R , и наоборот.

2

Рис. 2.35. Изменение ориентации листа на альбомную.

Вопросы для самопроверки

1. Перечислите основные виды команд LaTeX.
2. Для чего предназначены параметры и опции команды и в чем их отличие?
3. Какую роль выполняют окружения?
4. Перечислите базовые классы документов LaTeX, опишите их возможности.
5. Какая информация может быть включена в оформление колонтитулов?
6. В каком случае может потребоваться подключение дополнительных пакетов?
7. Как установить одинаковые поля для всей страницы, используя пакет `geometry`?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

Проработка примеров занятия

1. Изучите все примеры текущего занятия.
2. На каждый пример создайте TeX и PDF файл.
3. Скопируйте сделанное Вами задание «Математические методы в психологии» (практикум 4). Поменяйте поля листа на 2.5 см слева/справа, на 3 см сверху и 5 см снизу. Установите глобальный размер шрифта в 11pt. Сделайте двухколоночную разметку. Ожидаемый результат представлен на рис. 2.36:

Якубович Д.А.

15 апреля 2018 г.

1 Числовые характеристики распределений

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n} = \frac{1}{n} \sum_{k=1}^n X_k. \quad (1)$$

Символ суммы в формуле (1) используется для сокращения записи.

Мода — наиболее часто встречающееся числовое значение в выборке. Обозначается \bar{X} .
Например, среднее арифметическое первых десяти натуральных чисел равно

Например, среди чисел 1, 3, 2, 3, 4, 3, 6, 0 модой является число 3.

$$\overline{X} = \frac{1 + 2 + \dots + 10}{10} = 5.5. \quad (2)$$

1.2 Медиана

Медиана — значение, которое делит упорядоченное множество числовых данных пополам. Обозначается $Md(X)$. Например, для ряда 3, 4, 0, 2, 0, 5, 1, 1, 3 медиана $Md(X) = 2$.

1.4 Амплитуда ряда

Амплитудой числового ряда называют разницу между наибольшим и наименьшим значением: $A(X) = \max X - \min X$. Например, для числового ряда $X = 5, 0, -3, 1, 5, 2$ амплитуда равна $A(X) = 5 - (-3) = 8$.

1.3 Среднее арифметическое

Среднее арифметическое ряда из n числовых значений X_1, X_2, \dots, X_n вычисляется как

Рис. 2.36. Итоговый вид документа.

2.5 Шрифт

2.5.1 Настройка начертания шрифта

Высокое полиграфическое качество документа достигается, прежде всего, благодаря качественным шрифтам и тщательно подобранной их гарнитуре. LaTeX стремится одновременно оптимизировать компоновку элементов согласно классу документа и сохранить хорошее качество результата. При этом пользователю доступны команды для дополнительной настройки форматирования текста и его фрагментов.

Команды настройки начертания

Указанные ниже команды (рис. 2.37) меняют начертание шрифта для текста, помещенного в скобки команды. Макрос `\emph` является **семантическим**, т.е. он несет логический смысл «выделенный текст» и его форматирование различно в разных ситуациях. Остальные команды отвечают за физическое форматирование, т.е. определенное оформление указанного текста.

Команда	Вид
<code>\textbf{ }</code>	полужирный
<code>\textmd{ }</code>	средняя насыщенность
<code>\textrm{ }</code>	прямой
<code>\textup{ }</code>	прямой
<code>\textit{ }</code>	<i>курсив</i>
<code>\emph{ }</code>	<i>выделение</i>
<code>\textsl{ }</code>	<i>наклонный</i>
<code>\textsf{ }</code>	рубленный
<code>\textsc{ }</code>	КАПИТЕЛЬ
<code>\texttt{ }</code>	печатная машинка

Рис. 2.37. Команды настройки начертания шрифта.

Допускается вложение одной команды в другую, что позволяет применять несколько параметров форматирования одновременно.

Пример разметки LaTeX

```
\textbf{жирное начертание}
```

```

\textbf{\textit{жирный и курсивом}}

\textit{\textbf{курсивный и жиром =)}}

\textsc{ТЕКСТ капителью}

\textsc{ТЕКСТ \textbf{капителью и пожирнее}}

\texttt{имитация печатной машинки}

\textit{все курсивом} кроме \textit{одного слова}

```

Результат

жирное начертание
жирный и курсивом
курсивный и жиром =)
 ТЕКСТ КАПИТЕЛЬЮ
 ТЕКСТ КАПИТЕЛЬЮ И ПОЖИРНЕЕ
 имитация печатной машинки
все курсивом кроме *одного слова*

Рис. 2.38. Пример настройки начертания.

Декларации для настройки начертания

Каждой команде начертания соответствует **декларация**, оформляющая начертание абсолютно так же (рис. 2.38). Текст, что находится после декларации, меняет свое оформление.

Декларация	Вид
<code>\bfseries</code>	полужирный
<code>\mdseries</code>	средняя насыщенность
<code>\rmfamily</code>	прямой
<code>\sffamily</code>	рубленый
<code>\ttfamily</code>	печатная машинка
<code>\upshape</code>	прямой
<code>\itshape</code>	<i>курсив</i>
<code>\slshape</code>	<i>наклонный</i>
<code>\scshape</code>	КАПИТЕЛЬ

Рис. 2.39. Декларации для настройки начертания шрифта.

Декларация действует либо внутри скобок группы { }, либо пока не встретит другую «родственную» декларацию, которая меняет дальнейшее форматирование или дополняет его (т.е. формат предыдущей декларации сохраняется).

Пример разметки LaTeX

{\bfseries Декларации} преобразуют действия других команд или же создают другие команды. В этом примере декларации форматировуют текст.

Важно помнить, что область действия такой декларации можно

ограничить фигурными скобками группы \{ \}, как это сделано

для первого слова. Если же этого не сделать, то \itshape декларация форматировует текст до тех пор, пока не встретит \bfseries другую декларацию из своего семейства, причем форматирование предыдущей декларации может наследоваться.

% сброс курсива и полужирного начертания

\upshape \mdseries

Команды выше сбросили курсивное и жирное начертания.

Результат

Декларации преобразуют действия других команд или же создают другие команды. В этом примере декларации форматировуют текст.

Важно помнить, что область действия такой декларации можно ограничить фигурными скобками группы { }, как это сделано для первого слова. Если же этого не сделать, то декларация форматировует текст до тех пор, пока не встретит другую декларацию из своего семейства, причем форматирование предыдущей декларации может наследоваться.

Команды выше сбросили курсивное и жирное начертания.

Рис. 2.40. Пример настройки начертания.

Старайтесь использовать команды для небольшого текста (в несколько слов), а декларации – для оформления текста в несколько строк или абзацев.

Устаревшие декларации

Следующие декларации являются устаревшими, но до сих пор поддерживаются новыми версиями LaTeX:

Декларация	Вид
<code>\rm</code>	Романский стиль
<code>\it</code>	<i>Курсив</i>
<code>\bf</code>	Полужирный
<code>\sc</code>	КАПИТЕЛЬ
<code>\sf</code>	Рубленый
<code>\sl</code>	<i>Наклонный</i>
<code>\tt</code>	Печатная машинка

Рис. 2.41. Устаревшие декларации начертания.

В отличие от деклараций п. 0, для старых деклараций действие последующей декларации отменяет действие предыдущей, т.е. форматирование не наследуется.

Пример разметки LaTeX

Устаревшие `{\bf}` декларации все еще используются. Однако от такой практики постепенно отказываются.

Кроме того, `\it` устаревшие декларации не наследуют форматирование, а просто его `\bf` сбрасывают, поэтому комбинировать их смысла нет.

`\rm`

Таким образом, `\textbf{рекомендуется}` использовать новые команды и декларации.

Результат

Устаревшие **декларации** все еще используются. Однако от такой практики постепенно отказываются.

Кроме того, *устаревшие декларации не наследуют форматирование, а просто его сбрасывают, поэтому комбинировать их смысла нет.*

Таким образом, **рекомендуется** использовать новые команды и декларации.

Рис. 2.42. Пример использования устаревших деклараций.

Кегль

Настройка размера шрифта (**кегля**) в LaTeX имеет ряд особенностей.

Во-первых, основной размер шрифта определяется классом документа (берется по умолчанию, либо тот, который указан пользователем в опциях (10pt, 11pt или 12pt):

```
\documentclass[12pt]{book}
```

Во-вторых, размер фрагментов текста можно изменить, используя следующие декларации:

Декларация	Вид
<code>\tiny</code>	Крошечный
<code>\scriptsize</code>	Индексный
<code>\footnotesize</code>	Подстрочечный
<code>\small</code>	Маленький
<code>\normalsize</code>	Стандартный
<code>\large</code>	Увеличенный
<code>\Large</code>	Большой
<code>\LARGE</code>	Крупный
<code>\huge</code>	Огромный
<code>\Huge</code>	Громадный

Рис. 2.43. Декларации для изменения кегля шрифта.

При этом размер устанавливается относительно основного шрифта (что эквивалентно действию декларации `\normalsize`). Иными

словами, размер определяется пропорционально основному. Точные значения размера можно узнать в справочной литературе по LaTeX.

Пример разметки LaTeX

```
Нормальный размер шрифта устанавливается по умолчанию.  
\footnotesize Маленький шрифт,  
\normalsize нормальный шрифт,  
\Large большой шрифт.  
{\scriptsize Скобки группы ограничивают действие  
команды.}  
Шрифт всё еще большой  
\normalsize ... и снова нормальный
```

Результат

Нормальный размер шрифта устанавливается по умолчанию. Маленький шрифт, нормальный шрифт, большой шрифт. Скобки группы ограничивают действие команды. Шрифт всё еще БОЛЬШОЙ ... и снова нормальный

Рис. 2.44. Пример настройки кегля.

2.5.2 Настройка цвета

Прежде чем изменить цвет шрифта, нужно определить целесообразность подобного решения. Из курса психологии восприятия известно, что яркий цвет текста привлекает внимание, однако большой объем текста им лучше не оформлять. В работах учебного или научного характера цвет меняется достаточно редко (для выделения терминов, заголовков, цитат).

Тем не менее, LaTeX поддерживает разные цветовые модели и позволяет менять цвет текста, если это необходимо.

Пакеты `color` и `xcolor`

Пакет `color` позволяет работать с достаточно ограниченным набором цветов (их названия можно найти в специальных таблицах документации), например:

```
{\color{blue} Это пример текста синего цвета.}
```

Скобки группы необходимы, чтобы текст менял цвет только в указанном диапазоне.

Более мощным является пакет **xcolor**. Он позволяет работать с разными цветовыми моделями. В этом курсе мы ограничимся популярной трехкомпонентной RGB-схемой («красный-зеленый-голубой»). Для установки цвета используется команда `\textcolor`. Первым параметром команды является цвет, а вторым – текст (в опциях указана сама модель):

```
\textcolor{blue}{Это пример текста синего цвета.}
% Или произвольный цвет:
\textcolor[RGB]{45,20,240}{Это пример текста темно
синего цвета.}
```

Чтобы получить RGB-код цвета, можно воспользоваться любым приложением, содержащим работу с цветовой палитрой. Также существует множество веб-ресурсов, позволяющих работать с палитрой.

Например:

- https://www.w3schools.com/colors/colors_picker.asp;
- <https://colorscheme.ru/>;
- <https://paletton.com/>.

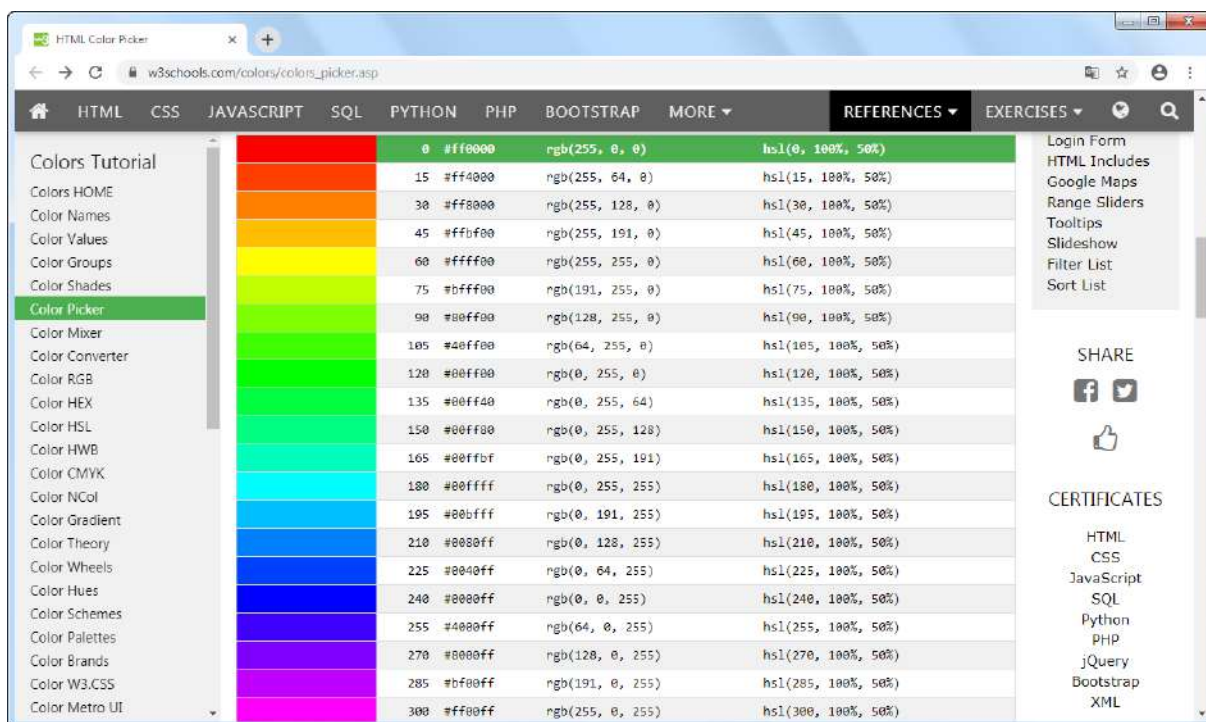


Рис. 2.45. Пример палитры цветов от сайта W3C.

Рекомендуемая практика установки цвета

Поскольку в работе несколько фрагментов текста могут быть оформлены одним цветом, то рациональнее ввести предварительное описание этого цвета. Для этого в преамбуле используется команда **\definecolor**, которая позволяет задать определенному цвету имя и далее его многократно использовать.

```
\definecolor{term}{RGB}{44,118,69}
\definecolor{comment}{RGB}{87,109,99}
```

Далее созданные цвета можно использовать уже как стандартные идентификаторы цвета.

В следующем примере приведены несколько приемов работы с цветом, рассмотренные в этом пункте.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{color}
\usepackage{xcolor}

% Определяем цвета
\definecolor{term}{RGB}{44,118,69}      % термин
\definecolor{comment}{RGB}{87,109,99}   % комментарий

\linespread{1.25}

\begin{document}
  \large

  {\color{blue} Это пример текста синего цвета.}

  \textcolor{blue}{Это пример текста синего цвета.}

  \textcolor{term}{\textbf{Информатика}} -- это наука
  о ...

  \textcolor{comment}{Это текст комментария.}

\end{document}
```

Результат

Это пример текста синего цвета.
Это пример текста синего цвета.
Информатика – это наука о ...
Это текст комментария.

Рис. 2.46. Пример настройки цвета шрифта.

Вопросы для самопроверки

1. Чем команда форматирования начертания отличается от декларации?
2. Почему устаревшие декларации форматирования начертания не следует использовать?
3. Почему декларации по изменению кегля шрифта называют относительными?
4. Какие цветовые модели поддерживает LaTeX?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Настройка гарнитуры и размера шрифта

Подготовьте документ с следующей разметкой:

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.25}

\begin{document}
  \part{Лекционный курс}
  \section{Информатика}
```

\subsection{Понятие информатики}

Информатика (от информация и автоматика) --- наука о методах и процессах сбора, хранения, обработки, передачи, анализа и оценки информации, обеспечивающих возможность её использования для принятия решений.

Информатика --- молодая научная дисциплина, изучающая вопросы, связанные с поиском, сбором, хранением, преобразованием и использованием информации в самых различных сферах человеческой деятельности.

Генетически информатика связана с вычислительной техникой, компьютерными системами и сетями, так как именно компьютеры позволяют порождать, хранить и автоматически перерабатывать информацию в таких количествах, что научный подход к информационным процессам становится одновременно необходимым и возможным.

До настоящего времени толкование термина <<информатика>> ещё не является установившимся и общепринятым. Обратимся к истории вопроса, восходящей ко времени появления электронных вычислительных машин.

\end{document}

Дополните этот документ таким образом, чтобы:

- геометрия страницы форматируется так: слева, справа, снизу – 3 см, сверху – 2 см;
- для первого абзаца увеличьте шрифт, также выделите термины жирным и курсивом;
- во втором абзаце размер шрифта вернуть к обычному;
- в третьем абзаце шрифт сделать рубленным и существенно уменьшить (см. рис. 2.48).

Продолжите документ следующим образом.

- После третьего абзаца добавьте код:
% Разрыв страницы:
\newpage
% Убрать отступы красной строки:
\parindent = 0cm

\subsection{Размеры}

- С помощью команд работы с кеглем получите результат, как на рис. 2.49.

2. Настройка цвета шрифта

Оформите следующий фрагмент текста (рис. 2.50). Используйте RGB-палитру цветов. Для этого введите в Google запрос «Color Picker» и подберите 10 оттенков зеленого цвета. Коды цветов описаны в окошечке RGB.

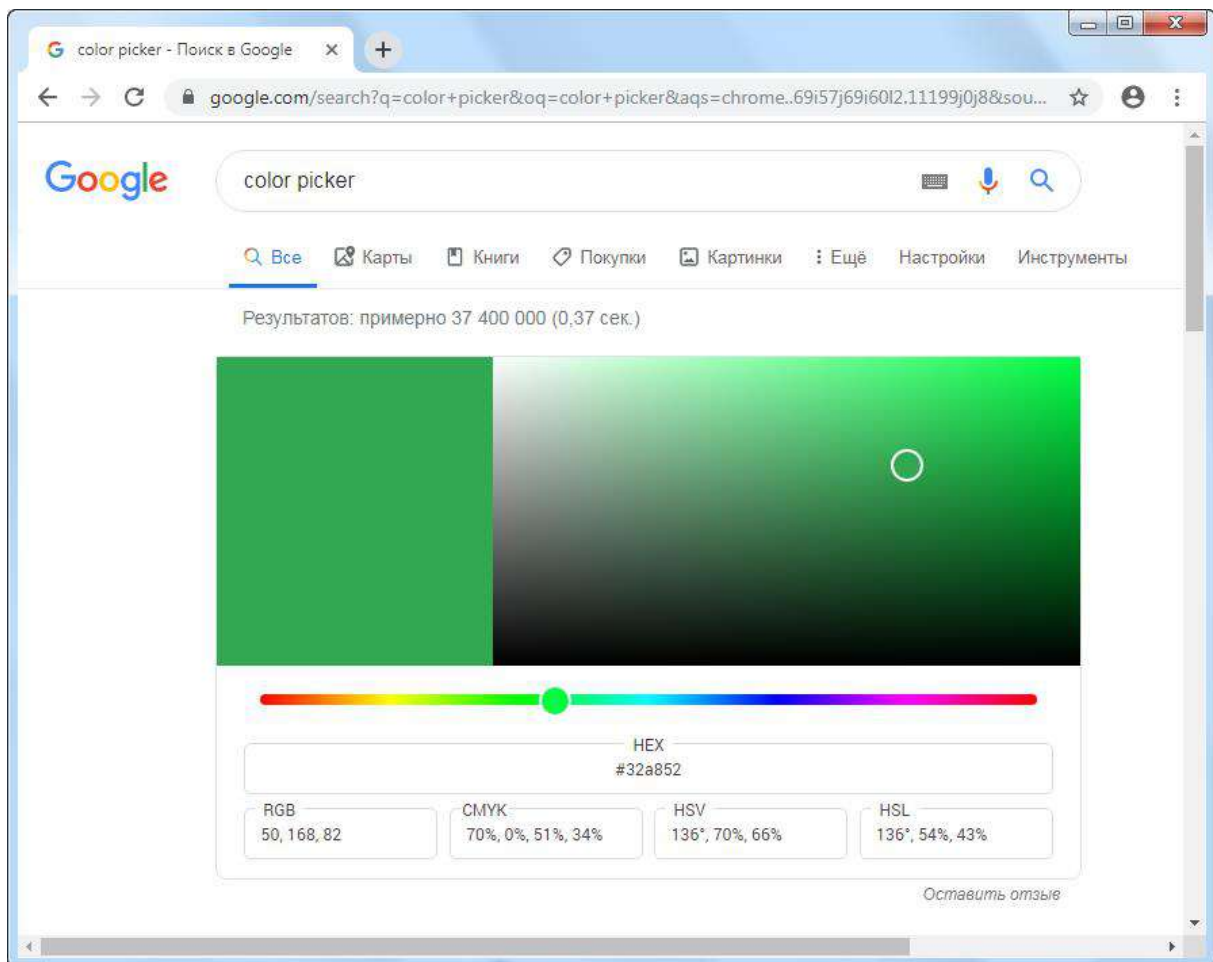


Рис. 2.47. Палитра цветов.

Результат 1

Часть I

Лекционный курс

1 Информатика

1.1 Понятие информатики

Информатика (от *информация* и *автоматика*) — наука о методах и процессах сбора, хранения, обработки, передачи, анализа и оценки информации, обеспечивающих возможность её использования для принятия решений.

Информатика — молодая научная дисциплина, изучающая вопросы, связанные с поиском, сбором, хранением, преобразованием и использованием информации в самых различных сферах человеческой деятельности. Генетически информатика связана с вычислительной техникой, компьютерными системами и сетями, так как именно компьютеры позволяют порождать, хранить и автоматически перерабатывать информацию в таких количествах, что научный подход к информационным процессам становится одновременно необходимым и возможным.

До настоящего времени толкование термина «информатика» ещё не является установившимся и общепринятым. Обратимся к истории вопроса, восходящей ко времени появления электронных вычислительных машин.

Рис. 2.48. Ожидаемый результат для задания 1.

Результат 2

1.2 Размеры

Тестовая строка 1

Тестовая строка 2

Тестовая строка 3

Тестовая строка 4

Тестовая строка 5

Тестовая строка 6

Тестовая строка 7

Тестовая строка 8

Тестовая строка 9

Тестовая строка 10

Рис. 2.49. Ожидаемый результат для задания 2.

Результат 3

Цвет текста 1
Цвет текста 2
Цвет текста 3
Цвет текста 4
Цвет текста 5
Цвет текста 6
Цвет текста 7
Цвет текста 8
Цвет текста 9
Цвет текста 10
Цвет текста 11

1

Рис. 2.50. Ожидаемый результат для задания 3.

2.6 Абзацы

2.6.1 Абзац в структуре документа

Абзац обозначает как логическое, так и структурное разделение текстовой информации на блоки, предполагающие завершенность некоторой идеи, мысли, сюжета. Разделение текста на абзацы:

- усиливает смысловой акцент фрагментов текста;
- способствует эффективному восприятию информации.

В зависимости от требований к оформлению, типа документа или средства просмотра (печатный / веб-документ) абзацы могут быть отделены отступом красной строки или дополнительным интервалом между абзацами.

Основываясь на ранее приведенных примерах, можно заметить, что

- любое число пробелов между словами в кодовой разметке документа всегда равносильно одному в итоговом;
- переход на новую строку не создает новый абзац, а пустая строка между блоками текста – создает.

Чтобы понять, по каким принципам LaTeX выделяет абзацы, обратимся к следующим пунктам.

Параметры абзаца

За разделение текста на абзацы отвечает следующая группа команд.

Команда	Описание
пустая строка	Пустая строка между текстовыми блоками равносильна началу нового абзаца. Несколько пустых строк равносильны одной пустой.
<code>\par</code>	Начало нового абзаца. Используется как альтернатива пустой строке.
<code>\noindent</code>	Подавляет отступ красной строки (если он по каким-либо причинам не нужен).
<code>\indent</code>	Ставит абзацный отступ (если он по каким-либо причинам не отобразился).

Разрыв строки

Разрыв строки не начинает новый абзац: его задача начать новую строку абзаца. Как правило, разрывы строки рекомендуется про- ставить уже в окончательно оформленном документе, в строках, где искусственно требуется избежать нежелательный перенос слова.

Команда	Описание
<code>\\</code>	Разрывает строку абзаца.
<code>*</code>	Разрывает строку абзаца, но избегает ее начала с новой страницы.
<code>\\[число]</code>	Разрывает строку и устанавливает указанное вертикальное расстояние между разорванными строками.
<code>*[число]</code>	Разрывает строку, но избегает ее начала с новой страницы; также устанавливает указанное вертикальное расстояние между разорванными строками.
<code>\linebreak</code>	Разрывает строку абзаца, причем последняя строка выравнивается по ширине (если оно используется).
<code>\-</code>	Указывает возможный перенос в слове (обычно используется для слов, не входящих в словари).
<code>~</code>	Пробел между словами (символами), которые запрещается разрывать (удобно указывать для ФИО, например Якубович~Д.А. и при указании перекрестных ссылок, чтобы номер не отрывался от подписи, например теорема~\ref{teor1}).

Разрыв страницы

Подобно MS Word, в LaTeX допускается возможность установ- ки разрыва страницы. Все, что указано после разрыва, будет начи- наться с новой страницы.

Команда	Описание
<code>\newpage</code>	Начинает новую страницу.
<code>\nopagebreak</code>	Запрещает разрыв страницы в указанном месте.

В следующем примере продемонстрировано комбинированное использование форматирования абзацев и абзацных отступов, а также используются разрывы строк.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.15}
\parindent = 1.25cm % Красная строка
\parskip = 0.5cm % Отступ между абзацами

\begin{document}
  \section{Абзацы}
  Абзацы и разрывы строк 1.
  Абзацы и разрывы строк 1.

  Абзацы и разрывы строк 2.
  \par
  Абзацы и разрывы строк 2.

  Абзацы \\\ и разрывы \\\ строк 3.

  \noindent
  Абзацы \\\ и разрывы \\\ строк 4.

  \noindent
  Абзацы \\\[0.25cm] и разрывы \\\[0.5cm] строк 5.
\end{document}
```

Результат

1 Абзацы

Абзацы и разрывы строк 1. Абзацы и разрывы строк 1.

Абзацы и разрывы строк 2.

Абзацы и разрывы строк 2.

Абзацы
и разрывы
строк 3.

Абзацы
и разрывы
строк 4.

Абзацы
и разрывы
строк 5.

Рис. 2.51. Настройка абзацев и разрывов строк.

Интерлиньяж

Интерлиньяж (междустрочный интервал) можно настроить как стандартными средствами, так и с помощью пакета **setspace**.

Способ 1

Предполагает переопределение действия команды **\baselinestretch**; например:

```
\renewcommand{\baselinestretch}{1.5}
```

Эта операция влияет только на глобальное изменение интервала.

Способ 2

Основан на работе с командой **\linespread**; например:

```
\linespread{1.5}
```

Возможности команды также ограничиваются глобальным интервалом.

Способ 3

Пакет `setspace` позволит установить интервал как для всего документа, так и отдельно для определенного фрагмента:

```
\usepackage{setspace}
\singlespacing      % одинарный интервал
\onehalfspacing     % полуторный интервал
\doublespacing      % двойной интервал
\setstretch{1.35}   % произвольный интервал
```

Если вам необходимо локально изменить интерлиньяж внутри блока, воспользуйтесь окружениями из этого пакета:

```
\begin{singlespace}
    Здесь одинарный интервал
\end{singlespace}

\begin{onehalfspace}
    Здесь полуторный интервал
\end{onehalfspace}

\begin{doublespace}
    Здесь двойной интервал
\end{doublespace}

\begin{spacing}{1.35}
    Здесь указанный интервал
\end{spacing}
```

Режимы выравнивания текста абзаца

По умолчанию в LaTeX принято выравнивание текста абзаца по ширине. Для смены режима выравнивания доступны три следующих окружения:

```
\begin{flushleft}
    Этот фрагмент текста выравнивается по левому краю.
\end{flushleft}

\begin{center}
    Этот фрагмент текста выравнивается по центру.
\end{center}
```

```
\begin{flushright}
```

Этот фрагмент текста выравнивается по правому краю.

```
\end{flushright}
```

Эти окружения также устанавливают дополнительные интервалы между абзацами (сверху и снизу).

В следующем примере демонстрируются режимы выравнивания и настройка интерлиньяжа. Кроме того, используется пакет **blindtext**. Он позволяет вставлять текст-заполнитель Lorem. Это очень распространенная практика в верстке документов и веб-страниц: можно быстро проверить форматирование на некотором абстрактном тексте, чтобы не тратить время на набор или копирование искомого. Команда **\blindtext** вставляет несколько предложений этого текста (переносы слов не предусмотрены, т.к. это цитата на латинском).

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Пакет для работы с текстом-заполнителем
\usepackage{blindtext}
% Пакет для настройки интерлиньяжа
\usepackage{setspace}
\linespread{1.25}

\begin{document}
  \section{Lorem}
  \blindtext

  \begin{center}
    \begin{spacing}{1.75}
      \blindtext
    \end{spacing}

    \begin{spacing}{0.7}
      Текстовый абзац 1.

      Текстовый абзац 2.
    \end{spacing}
  \end{center}
\end{document}
```


Результат

1 Lorem

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum.

Nunc quis urna dictum turpis accumsan semper.

Текстовый абзац 1.
Текстовый абзац 2.

Рис. 2.52. Настройка интерлиньяжа и выравнивания.

В примере

- первый абзац имеет интерлиньяж 1.25;
- второй, третий и четвертый абзацы выравниваются по центру;
- у второго абзаца интерлиньяж равен 1.75;
- третий и четвертый абзацы получают интервал по 0.7.

2.6.2 Вертикальные и горизонтальные отступы

Читатель уже наверняка смог заметить, что любое число пробелов в разметке текста равносильно одному пробелу. В LaTeX настройка горизонтальных и вертикальных пробелов осуществляется специальными командами.

Горизонтальные отступы и пробелы

Команда	Описание
<code>\,</code>	Очень маленький пробел.
<code>\</code> (слеш с пробелом)	Обычный пробел.
<code>\:</code>	Средний пробел. (Работает только в формулах!)
<code>\;</code>	Толстый пробел. (Работает только в формулах!)
<code>\!</code>	Тонкий отрицательный пробел. (Работает только в формулах!)
<code>\/</code>	Корректирующий пробел (полезен при переходе от наклонного (курсивного) шрифта к нормальному или наоборот).
<code>\hspace{n}</code>	Пробел длины n. Если находится с краю строки, то удаляется.
<code>\hspace*{n}</code>	Пробел длины n. Не удаляется, если находится с краю строки.

Вертикальные отступы

Команда	Описание
<code>\vspace{n}</code>	Вертикальный отступ длины n. Если стоит в начале или в конце страницы, то убирается.
<code>\vspace*{n}</code>	Вертикальный отступ длины n. Не убирается, если стоит в начале или в конце страницы.

Посмотрим, как вертикальные и горизонтальные пробелы влияют на форматирование текста.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.25}

\begin{document}
  \large
  Маленький\, пробел

  Обычный\ пробел

  Обычный пробел

  Настраиваемый\hspace{1.5cm}пробел

  Настраиваемый\hspace{1.5cm} пробел

  Настраиваемый \hspace{1.5cm}пробел

  Настраиваемый \hspace{1.5cm} пробел

  \vspace{5cm}
  Вертикальный отступ от предыдущего абзаца на 5 см.
  \vspace{5cm}
  Поскольку новый абзац не начат, то это предложение
  начнет печататься в текущем абзаце, а вторая и
  последующие строки будут смещены на указанный
  интервал. Подобного следует избегать.
\end{document}
```

Результат

Маленький пробел	
Обычный пробел	
Обычный пробел	
Настраиваемый	пробел
Настраиваемый	пробел
Настраиваемый	пробел
Настраиваемый	пробел

Вертикальный отступ от предыдущего абзаца на 5 см. Поскольку

новый абзац не начат, то это предложение начнет печататься в текущем абзаце, а вторая и последующие строки будут смещены на указанный интервал. Подобного следует избегать.

Рис. 2.53. Пример настройки пробелов и отступов.

Обратите внимание (см. рис. 2.53): обычный пробел в разметке добавляется к горизонтальному!

2.6.3 Заполнители

В LaTeX предусмотрены специальные команды, называемые **заполнителями**. Заполнитель как-бы «расталкивает» два элемента относительно левой / правой либо верхней / нижней границ листа. Горизонтальный заполнитель может быть оформлен также в виде точек или линии. Заполнители можно вставлять и между более чем двумя объектами: в этом случае система постарается заполнить пространство между ними, сохранив пропорции.

Заполнители в виде пустого пространства серьезно упрощают процесс компоновки элементов на странице: не нужно заботиться о настройке отступов и разрывов страницы.

Горизонтальный заполнитель

Команда	Описание
<code>\hfill</code>	Бесконечно растяжимый пробел между элементами.
<code>\hrulefill</code>	Бесконечно растяжимая линия заполнитель между элементами.
<code>\dotfill</code>	Бесконечно растяжимый заполнитель точками между элементами.

Все три типа заполнителя схематично представлены на рис. 2.54.

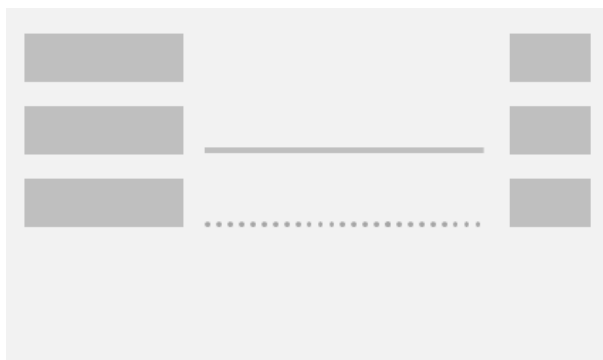


Рис. 2.54. Виды горизонтальных заполнителей.

Вертикальный заполнитель

Команда	Описание
<code>\vfill</code>	Бесконечно растяжимый вертикальный пробел между двумя элементами.

Схематично вертикальный заполнитель расталкивает содержимое страницы, как изображено на схеме рис. 2.55.

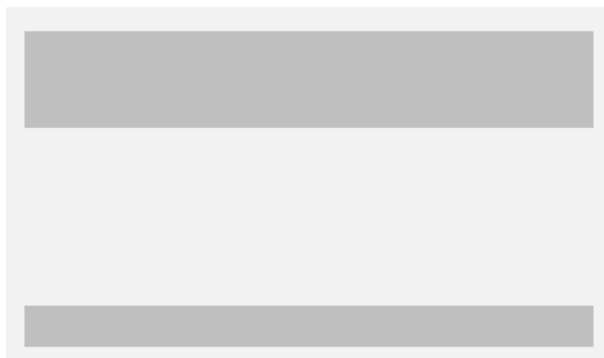


Рис. 2.55. Вертикальный заполнитель.

Приведем пример использования разного типа заполнителей. Горизонтальный заполнитель в форме линии прекрасно подойдет для оформления строки для написания текста. Пустой заполнитель будет расталкивать содержимое горизонтально и вертикально. А следовательно, не нужно будет рассчитывать величину отступа и заботиться о том, что текст сместится на новую страницу.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Убираем нумерацию страниц
\pagestyle{empty}

\linespread{1.05}

% Убираем отступы красной строки
\parindent = 0cm
```

```

\begin{document}
  % Задаем крупный размер шрифта
  \large

  % Оформляем первую строку
  Ф.И.О. \hrulefill

  % Оформляем вторую строку
  \vspace{2em}
  Дата доклада: \hrulefill\ Город: \hrulefill

  % Оформляем линии
  \vspace{2em}
  Вопросы доклада: \hrulefill \par
  1. \hrulefill \par
     \hrulefill \par
  2. \hrulefill \par
     \hrulefill \par

  % Прижимаем текст ниже к нижнему краю листа
  \vfill
  % Оформляем реквизиты
  Отчет проверен \\\
  зав. кафедры ИИТО, \\\
  д.ф.-м.н., профессором
  \hfill
  Ивановым И.И.
\end{document}

```

Ф.И.О. _____

Дата доклада: _____ Город: _____

Вопросы доклада: _____

1. _____

2. _____

Отчет проверен
зав. кафедры ИИТО,
д.ф.-м.н., профессором

Ивановым И.И.

Вопросы для самопроверки

1. Укажите способы, с помощью которых можно начать новый абзац.
2. В чем разница между разрывом строки и началом нового абзаца?
3. Какими командами можно настроить интерлиньяж во всем документе и в определенном его месте?
4. Перечислите команды для установки пробелов между словами и интервалов между соседними абзацами.
5. Приведите примеры использования заполнителей в документе.

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Закрепление пройденного материала

1. Раскройте основные отличия команд начала нового абзаца от разрыва абзаца.
2. Воспроизведите примеры из текущего занятия.
3. В ряде примеров измените числовые значения отступов, интервалов. Проследите за поведением элементов.

2. Форматирование абзацев

1. Оформите документ (см. рис. 2.57), как указано в его тексте.
2. Используя команды для работы с абзацами, выравниванием, установку вертикальных отступов, геометрии страницы, а также настройки размера и начертания шрифта, оформите объявление, как на рис. 2.58. Замечание: установите в преамбуле `\parindent = 0cm` (сбросить отступ красной строки).
3. Используя заполнители, а также известные вам команды форматирования текста и абзацев, вертикальных отступов, создайте следующий бланк, изображенный на рис. 2.59. Поля документа: по 2 см сверху, слева, справа, и 3 см – снизу.

Результат 1

Междустрочный интервал во всем документе равен 1.3, отступ красной строки – 0.75 см.

Это текст абзаца 2.

Это текст абзаца 3.

Это текст абзаца 4. Для него осуществить подавление отступа красной строки.

Между этим и предыдущим абзацем установить вертикальный отступ в 1.5 см.

В этом
абзаце требуется
установить разрыв
через каждые
два слова.

Это предложение должно быть
отображено именно
так!

Для этого предложения
уменьшите междустрочный интервал
до 1.0.

Посреди этого предложения установили большой пробел.

Рис. 2.57. Ожидаемый результат задания 1.

Результат 2

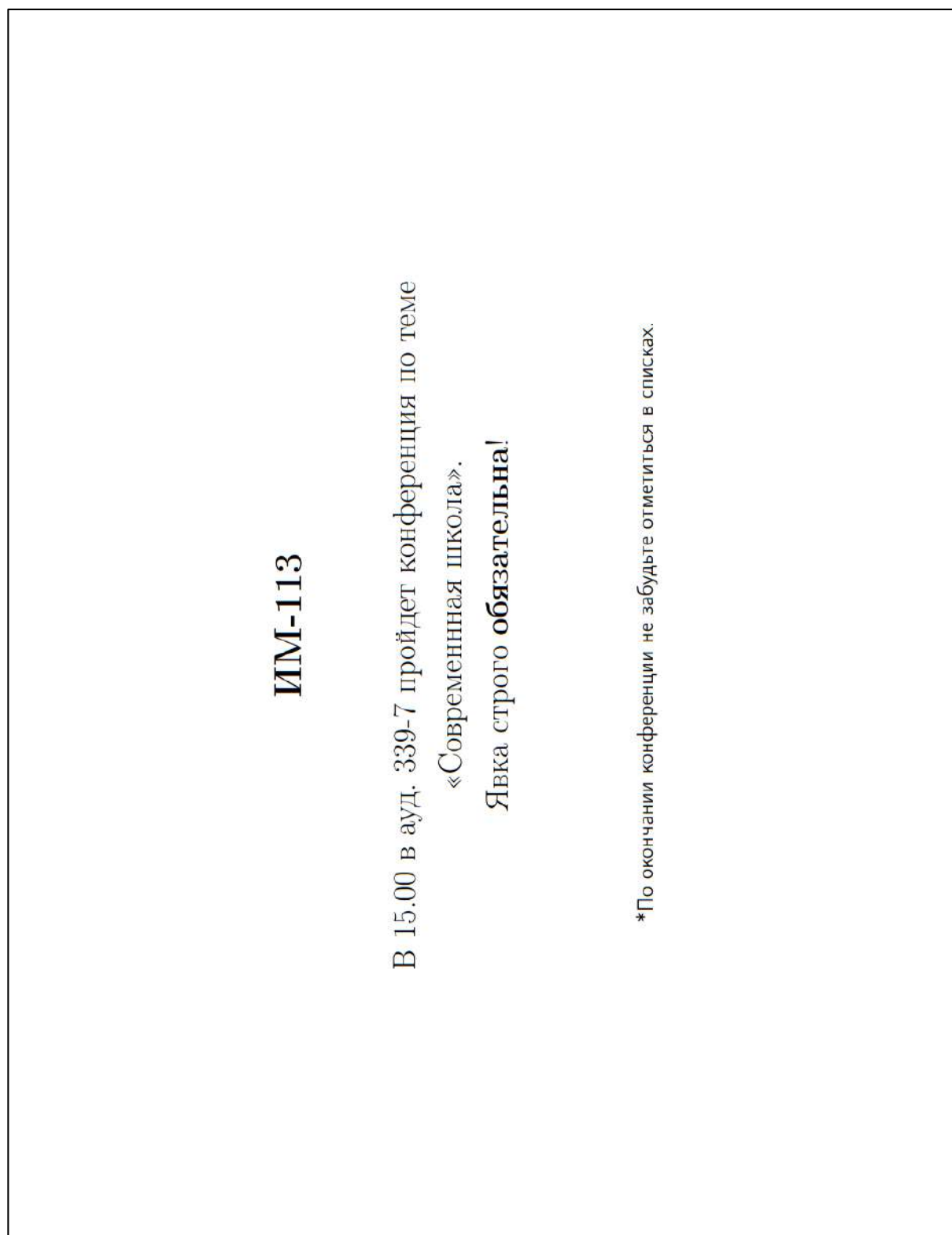


Рис. 2.58. Ожидаемый результат задания 2.

Анкета

Ф.И.О.: _____
Дата рождения: _____
Город: _____
Институт / университет: _____

Аннотация к докладу: _____

Автор работы

Автор А.А.

Научный руководитель
магистранта / аспиранта

Руководитель Р.Р.

Дата

Подпись _____

Рис. 2.59. Ожидаемый результат задания 3.

3. Общая компоновка текста

Создайте новый документ и скопируйте в него следующий код:

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Общие свойства форматирования текста документа

\begin{document}

    % Здесь будет текст

\end{document}
```

Осуществите следующее:

1. Скопируйте некоторый фрагмент учебного текста порядка 3-5 страниц (для простоты не берите текст с изображениями, таблицами и др.). Можете взять текст из википедии. Главное, чтобы в тексте можно было выделить секции и подсекции. Необходимо будет посмотреть, как LaTeX самостоятельно его настроит согласно базовым параметрам.
2. Разбейте текст на абзацы (согласно источнику).
3. Выделите жирным начертанием важные термины и курсивом менее важные.
4. Установите размер шрифта во всем документе 12pt, отступ красной строки 1.25 см, междустрочный интервал – 1.35.
5. С помощью команд рубрикации \section и \subsection разделите заголовки и подзаголовки.

В результате всех операций должен получиться документ, например, следующего вида (рис. 2.60-рис. 2.64).

Результат 4

1 Введение

Владимир — город в России, административный центр Владимирской области и городского округа город Владимир. В XII—XIV веках — столица Великого княжества Владимирского. Один из крупнейших туристических центров европейской части России. Входит в Золотое кольцо России.

2 История

2.1 Предыстория

Первые люди стали появляться на территории, которую занимает современный город, около 30-25 тысяч лет назад в эпоху палеолита, о чём свидетельствует открытие на правом берегу реки Рпень стоянки Русаниха, каменные орудия труда которой схожи с теми, что были найдены на стоянке Сунгирь, расположенной в 8 км к востоку от Русанихи.

В VI—VII веках эту территорию осваивало волго-финское племя меря. В VIII—X веках на холме, где позже был построен Успенский собор, существовал мерянский посёлок.

В IX—X веках сюда стали проникать славяне, в частности, кривичи.

2.2 Основание города

Раньше считалось, что Владимир был основан князем Владимиром Мономахом около 1108 года. Соответственно, 850-летие его отмечалось в

Рис. 2.60. Ожидаемый результат (стр. 1).

1958 году. Этой датировки придерживался и виднейший специалист по владимирской археологии Николай Воронин.

В качестве обоснования датировки основания Владимира 1108 годом приводилось сообщение Новгородской первой летописи: «Сын Володимиров Мономах, правнук великого князя Владимира. Сии поставил град Володимерь Залешьский в Суждальской земле и осыпал его спом, и созда первую церковь святого Спаса за 50 лет до Богородичина ставления».

Приводились также косвенные доказательства, связанные с тем, что ростовские и суздальские бояре смотрели на Владимир как на младший город и противодействовали его возвышению, между тем Суздаль впервые упомянут в Новгородском кодексе под 999 годом. Но претензии на «старшинство» могли быть связаны не с датой основания, а со множеством других более значимых в то время факторов.

В 1990-х годах владимирские краеведы высказались за перенос даты основания города на 990 год. В обоснование этой даты приводят-ся известия ряда летописных источников — Супрасльской, Густынской, Львовской, Ермолинской, Никоновской, Холмогорской летописей, сокращённых летописных сводов 1493 и 1495 годов, летописных сводов 1497 и 1518 годов, Степенной книги, Русского хронографа и т. п. Во всех этих текстах указывается, что Владимир-на-Клязьме основал Владимир Святославич в 990 году. Альтернативную датировку основания Владимира в то время поддержал академик Дмитрий Лихачёв.

В 2016 году исследователь владими́ро-суздальского зодчества, академик Сергей Заграевский опубликовал исследование «Историческая то-

пография домонгольского Владимира», в котором свёл воедино известные летописные источники и показал, что Владимир мог быть основан в 990 году. Эта дата была поддержана администрацией города Владимира.

2.3 Стольный град

Владимир начинает расти и усиливаться благодаря заботам Владимира Мономаха и Юрия Долгорукого, которые укрепляли его как опорный пункт защиты Ростово-Суздальского княжества. Расцветом обязан князю Андрею Боголюбскому, который в 1157 году перенёс сюда столицу княжества. Его преемник — Всеволод Большое Гнездо — был одним из самых могущественных русских князей, и с его правления за владимирскими князьями закрепляется титул «великих».

Во Владимире и соседнем Суздале сложилась владими́ро-сузда́льская школа живописи; в городе велось летописание.

Удар по Владимиру был нанесён в 1238 году нашествием монголо-татар (см. оборона Владимира). Позднее город несколько раз подвергался татарским набегам, самым тяжёлым из которых стало разорение Дюденевой ратью в 1293 году. Начиная с Василия Ярославича, в условиях усилившейся раздробленности в Северо-Восточной Руси Владимир перестал быть резиденцией великих князей, которые теперь только совершали в нём обряд интронизации, оставаясь жить в своих наследственных землях. Тем не менее именно владимирские князья были признаны в Орде старейшими во всей Руси, благодаря чему Владимир стал номинальной столицей русских земель. Этот статус ещё более укрепился в 1299 году, когда город стал резиденцией русских митрополитов.

Рис. 2.62. Ожидаемый результат (стр. 3).

2.4 Борьба за великое княжение Владимирское, Владимир и Москва

Как главный стол Северо-Восточной Руси, Владимир не имел собственной княжеской династии. Контроль над ним вместе с обширной доменной территорией получал по выдаваемому в Орде ханскому ярлыку один из удельных князей. На протяжении 1-й половины XIV века основными претендентами на великое княжение были Тверь, Москва и Суздаль. Важной вехой стал переезд в 1325 году из Владимира в Москву (при Иване Калите) митрополита Петра. Дмитрию Донскому удалось добиться признания наследственных прав на Владимир со стороны всех соседних князей и Орды, что означало слияние Московского и Владимирского княжеств.

В 1382 году Владимир, как и другие города Северо-Восточной Руси, пострадал от нашествия Тохтамыша. Во время нашествия Тамерлана в 1395 году чудотворная и особо чтимая икона Владимирской Богоматери была перенесена в Москву для защиты города от завоевателя. То, что войска Тамерлана без видимых причин повернули от Ельца обратно, не дойдя до Москвы, было расценено как заступничество Богородицы и икону обратно во Владимир уже не возвращали.

В 1408 году Владимирский Успенский собор был заново расписан артелью мастеров, в состав которой входили Андрей Рублёв и Даниил Чёрный. Считается, что помимо фресок, мастера также создали иконы для соборного монументального иконостаса, ставшего важным этапом формирования системы высокого русского иконостаса. В том же году город был разорён татарами и нижегородцами. В 1441 году Владимир

Рис. 2.63. Ожидаемый результат (стр. 4).

подвергся нападению татарского царевича Талыча и нижегородского воеводы Карамышева.

С ростом Московского государства Владимир переходит в разряд рядового провинциального города. Хотя в титулатуре русских князей и царей он по-прежнему ставился на первое место, впереди Москвы. В 1565 году после разделения царём Иваном Грозным Русского государства на опричнину и земщину, город вошёл в состав последней.

В 1614 году окрестности города были разорены войсками литовского авантюриста Лисовского.

Рис. 2.64. Ожидаемый результат (стр. 5).

Глава 3

МЕХАНИЗМЫ АВТОМАТИЗАЦИИ ОФОРМЛЕНИЯ И РАБОТА С ОБЪЕКТАМИ В LATEX

3.1 Рубрикация и оглавление

3.1.1 Команды рубрикации

Рубрика в LaTeX

Рубрикация – это разбиение текста на главы, разделы, параграфы и пункты.

В концепции LaTeX рубрика (секция) рассматривается в качестве структурной единицы, влияющей на блок оглавления и колонтитулов. Между рубриками существует четкое правило иерархии и они автоматически нумеруются с учетом вложения. Оформление рубрики зависит от ее места в иерархии (оно может быть переформатировано, если требуется).

Для каждого класса документа доступны свои уровни рубрикации. Раздел `\part` (часть) формально является старшим, но не является обязательным и не влияет на нумерацию младших разделов. Так, если последней в Части 1 была Глава 3, то Часть 2 продолжится с Главы 4. Поэтому команда `\chapter` (глава) считается главным уровнем в книгах. В статьях же, напротив – нет деления на главы, а самой старшей рубрикой является `\section` (секция, пункт).

Список команд рубрикации

Следующая таблица отображает команды рубрикации LaTeX. Колонки Article и Book отвечают за одноименные классы и демонстрируют, поддерживает ли класс соответствующую рубрику.

Команда	Название	Уровень	Класс	
			Article	Book
<code>\part{ }</code>	Часть	-1	+	+
<code>\chapter{ }</code>	Глава	0	-	+
<code>\section{ }</code>	Раздел	1	+	+
<code>\subsection{ }</code>	Подраздел	2	+	+
<code>\subsubsection{ }</code>	Подподраздел	3	+	+
<code>\paragraph{ }</code>	Параграф	4	+	+
<code>\subparagraph{ }</code>	Подпараграф	5	+	+

Следующая форма команды справедлива для любой рубрики:

`\chapter[краткий_заголовок]{заголовок}`

Команда определяет главу с указанным текстом заголовка.

- *Указанный заголовок будет отображаться в колонтитуле.*
- *В необязательной опции указывают краткий текст заголовка, который также включается в оглавление и колонтитул вместо полного.*

Следующий пример демонстрирует мощь LaTeX на достаточно большом фрагменте текста. Для краткости здесь указаны лишь некоторые заголовки. Обратите внимание, как формируются колонтитулы и нумерация страниц.

Пример

```

\documentclass[12pt]{book}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Задаем информация об авторе
\title{\textbf{\textsc{Курс лекций по психологии}}}
\author{ru.wikipedia.org}
\date{} % Без даты

\linespread{1.3}

```

```

\begin{document}
  % Сформировать титул
  \maketitle

  % ГЛАВА 1
  \chapter[Поведение]{Психология поведения}
  \section{Введение}
  \textbf{Поведение} --- определённый сложившийся ...

  \section{Подходы и направления}
  Корни наук о поведении уходят в античные ...

  \subsection{Психология}
  Психология изучает психику человека, ...
  \subsection{Зоопсихология}
  Зоопсихология занимается изучением ...

  . . .

  % ГЛАВА 2
  \chapter[Мышление]{Психология мышления}
  \section{Введение}
  \textbf{Мышление} --- это познавательная ...

  . . .

\end{document}

```

Результат

Приведем лишь несколько страниц в предположении, что читателю понятна общая идея (см. рис. 3.1-рис. 3.3).

КУРС ЛЕКЦИЙ ПО ПСИХОЛОГИИ

ru.wikipedia.org

Рис. 3.1. Пример автоматической рубрикации книги (оформление титула).

Глава 1

Психология поведения

1.1 Введение

Поведение — определённый сложившийся образ взаимодействия с окружающей средой. Поведение определяется способностью изменять свои действия под влиянием внутренних и внешних факторов.

Поведение — характерная черта животного типа организации, оно имеет огромное приспособительное значение, позволяя животным избегать негативных факторов окружающей среды; хотя поведение характерно и для более простых организмов, например, простейшие проявляют способность перемещаться в ответ на раздражители среды и способны к элементарным формам научения. У многоклеточных организмов поведение находится под контролем нервной системы. В целом можно отметить, что поведение возникает на высоком уровне организации, когда организм приобретает способность воспринимать, хранить и преобразовывать информацию, используя её с целью самосохранения и приспособления к условиям существования.

Поведение, в отличие от психики, доступно для непосредственного наблюдения и является предметом широкого спектра наук, от психологии,

Рис. 3.2. Пример автоматической рубрикации книги (оформление начала главы).

1.2.2 Зоопсихология

Зоопсихология занимается изучением психической деятельности животных. Объектом зоопсихологии является поведение животных. Предметом зоопсихологии являются особенности и закономерности психической деятельности животных, а также её развитие в онтогенезе и филогенезе. Внимание зоопсихологов направлено на изучение восприятия, памяти, мышления животных.

1.2.3 Поведение растений

Хотя растения и бактерии обладают способностью к перемещению под действием внешних факторов (таксису), а высшие растения также не лишены способности к движению и обучению, тем не менее, поскольку механизмы движения растений носят чисто физиологический характер, нельзя говорить о наличии у них психики. В психологии движения растений часто относят к «допсихическому уровню отражения».

1.2.4 Сравнительная психология

Сравнительная психология занимается сравнительным анализом психических процессов представителей разных таксономических групп. Характерной чертой сравнительной психологии является применение сравнительного анализа в качестве основного метода.

1.2.5 Бихевиоризм

Бихевиоризм — направление в американской психологии, зародившееся в начале XX века. Особенностью этого направления был отказ от изучения психики, как непознаваемого явления. При этом полностью исключались из рассмотрения психические процессы, а поведение сводилось к

Рис. 3.3. Пример автоматической рубрикации книги (оформление колонтитулов в зависимости от левого и правого разворота).

Команды *-рубрикации

Следующая форма команды справедлива для любой рубрики:

`\chapter*{заголовок}`

*Команда определяет главу с указанным текстом заголовка. В отличие от основной команды, *-заголовок не включается в нумерацию, не отображается в колонтитулах и содержании.*

Указанный тип рубрик полезен для оформления введения, заключения и других разделов документа, которые не следует нумеровать. Недостаток состоит в том, что *-рубрика не включается в оглавление. Однако далее мы рассмотрим прием включения *-рубрик в оглавление.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}

    % Автоматическая генерация оглавления (см. далее)
    \tableofcontents

    \section[Секция 1]{Название секции 1}
    \subsection*{Введение}
    . . .
    \subsection{Подзаголовок 1}
    . . .
    \subsection{Подзаголовок 2}
    . . .
    \subsection{Подзаголовок 3}
    . . .
    \subsection*{Заключение}
    . . .

\end{document}
```

Результат

Содержание	
1 Секция 1	1
1.1 Подзаголовок 1	1
1.2 Подзаголовок 2	1
1.3 Подзаголовок 3	1
 1 Название секции 1	
Введение	
...	
1.1 Подзаголовок 1	
...	
1.2 Подзаголовок 2	
...	
1.3 Подзаголовок 3	
...	
Заключение	
...	

Рис. 3.4. Пример использования команды *- рубрикации.

3.1.2 Генерация оглавления

Оглавление для обычных рубрик

Пользователям офисных приложений известно, что генерация оглавления к документу требует предварительной настройки и форматирования. Например, в MS Word существует специальный механизм сборки оглавления. В целом он достаточно прост и требует лишь грамотной настройки стилей заголовков.

В LaTeX задача генерации оглавления решается чрезвычайно элегантно.

`\tableofcontents`

Команда автоматически генерирует оглавление на основе созданных рубрик.

- *Если у рубрики указан краткий заголовок, то он и печатается.*
- *Рубрики с пометкой * не включаются в оглавление.*
- *Команда может находиться в любом месте документа (но рекомендуется помещать ее в самом начале, либо в конце документа).*

Включение *-рубрик в оглавление

Если вам необходимо включить *-рубрику в оглавление, используйте команду `\addcontentsline` (у нее 3 параметра):

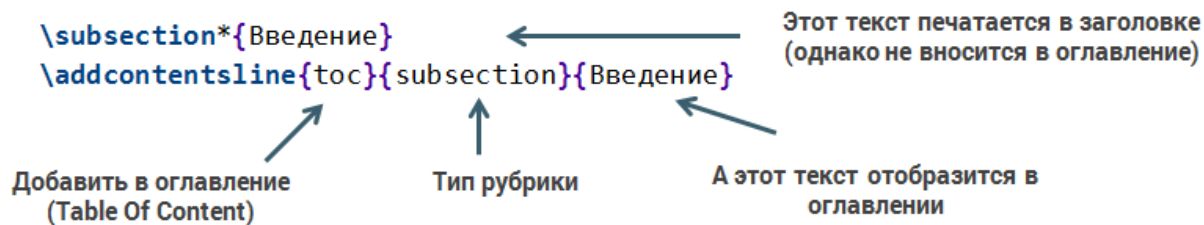


Рис. 3.5. Включение *-рубрики в оглавление.

В следующем примере введение и заключение включены в содержание (см. рис. 3.6):

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}
  % Автоматическая генерация оглавления
  \tableofcontents

  \section[Секция 1]{Название секции 1}
  \subsection*{Введение}
  \addcontentsline{toc}{subsection}{Введение}
  . . .
  \subsection{Подзаголовок 1}
  . . .
  \subsection{Подзаголовок 2}
  . . .
  \subsection{Подзаголовок 3}
  . . .
  \subsection*{Заключение}
  \addcontentsline{toc}{subsection}{Заключение}
  . . .

\end{document}
```

Результат

Содержание

1	Секция 1	1
	Введение	1
1.1	Подзаголовок 1	1
1.2	Подзаголовок 2	1
1.3	Подзаголовок 3	1
	Заключение	1

1 Название секции 1

Введение

• • •

1.1 Подзаголовок 1

• • •

1.2 Подзаголовок 2

• • •

1.3 Подзаголовок 3

• • •

Заключение

• • •

Рис. 3.6. Пример генерации оглавления.

3.1.3 Оформление приложений

`\appendix`

Декларация переключает нумерацию последующих рубрик в режим нумерации приложений.

- Разделы нумеруются латинскими символами A, B, C и т.д., подразделы – A.1, A.2 и т.д.
- Основной раздел указывается с ключевым словом «Приложение»: «Приложение A», «Приложение B» и т.д.
- Приложения также вносятся в оглавление.

Отметим, что *-рубрики для блока приложений сохраняют свои принципы оформления.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}
. . .

\appendix
\section{Заголовок приложения 1}
. . .
\subsection{Подзаголовок приложения 1}
. . .
\subsection{Подзаголовок приложения 2}
. . .

\section{Заголовок приложения 2}
. . .

\end{document}
```

Результат

Содержание	
1 Секция 1	2
Введение	2
1.1 Подзаголовок 1	3
1.2 Подзаголовок 2	4
1.3 Подзаголовок 3	5
Введение	6
A Заголовок приложения 1	6
A.1 Подзаголовок приложения 1	6
A.2 Подзаголовок приложения 2	6
B Заголовок приложения 2	6

A Заголовок приложения 1
...
A.1 Подзаголовок приложения 1
...
A.2 Подзаголовок приложения 2
...
B Заголовок приложения 2
...

Рис. 3.7. Автоматическое оформление приложений в оглавлении и в тексте.

Вопросы для самопроверки

1. Для чего предназначены команды рубрикации? Перечислите их.
2. В чем преимущество заголовков, оформленных с помощью команд рубрикации?
3. Как сгенерировать оглавление?
4. Зачем нужны *-рубрики и каковы их особенности?
5. Опишите команду \appendix.

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Закрепление пройденного материала

Изучите код разметки документа и его итоговый вид (текст сокращен, полностью его можно найти на сайте Википедии: <https://ru.wikipedia.org/wiki/Поведение>):

```
\documentclass[12pt]{book}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\title{\textbf{\textsc{Курс лекций по психологии}}\}
\author{ru.wikipedia.org}
\date{} % Без даты

\linespread{1.3}

\begin{document}
  \maketitle

  \chapter[Поведение]{Психология поведения}
  \section{Введение}
  \textbf{Поведение} --- определённый сложившийся образ
  взаимодействия с окружающей средой...

  \section{Подходы и направления}
```


Корни наук о поведении уходят в античные времена...

\subsection{Психология}

Психология изучает психику человека, а именно закономерности, особенности и развитие психической деятельности человека...

\subsection{Зоопсихология}

Зоопсихология занимается изучением психической деятельности животных...

\subsection{Поведение растений}

Хотя растения и бактерии ...

\subsection{Сравнительная психология}

Сравнительная психология занимается ...

\subsection{Бихевиоризм}

Бихевиоризм --- направление в американской психологии, зародившееся в начале XX века...

\subsection{Этология}

Этология в современном понимании --- наука о ...

\chapter[Мышление]{Психология мышления}

\section{Введение}

\textbf{Мышление} --- это познавательная деятельность человека...

\section{История представлений о мышлении}

Уже в античной науке с мышлением была связана ...

\end{document}

С помощью *-рубрикации сделайте так, чтобы заголовки с введением (для каждой главы) не нумеровались. Кроме того, пункты с номером 1 должны начинаться с новой страницы (рис. 3.8).

2. Генерация оглавления

Основываясь на тексте предыдущего примера, сгенерируйте к документу оглавление (см. рис. 3.9). Разделы с введением также должны быть включены в оглавление.

Отдельным документом сгенерировать оглавление в конце документа.



Рис. 3.8. Примеры страниц документа из задания 1.

Оглавление

1	Поведение	5
	Введение	5
1.1	Подходы и направления	7
1.1.1	Психология	7
1.1.2	Зоопсихология	7
1.1.3	Поведение растений	8
1.1.4	Сравнительная психология	8
1.1.5	Бихевиоризм	8
1.1.6	Этология	9
2	Мышление	11
	Введение	11
2.1	История представлений о мышлении	13

Рис. 3.9. Требуемый вид оглавления для задания 2.

3. Комбинирование на конкретном примере

Перейдите на страницу Wikipedia.org с описанием полиграфии: <https://ru.wikipedia.org/wiki/Полиграфия>.

Используя текст этой страницы, а также навыки работы с текстом и рубриками, реализуйте книгу следующего вида (рис. 3.10-рис. 3.13):

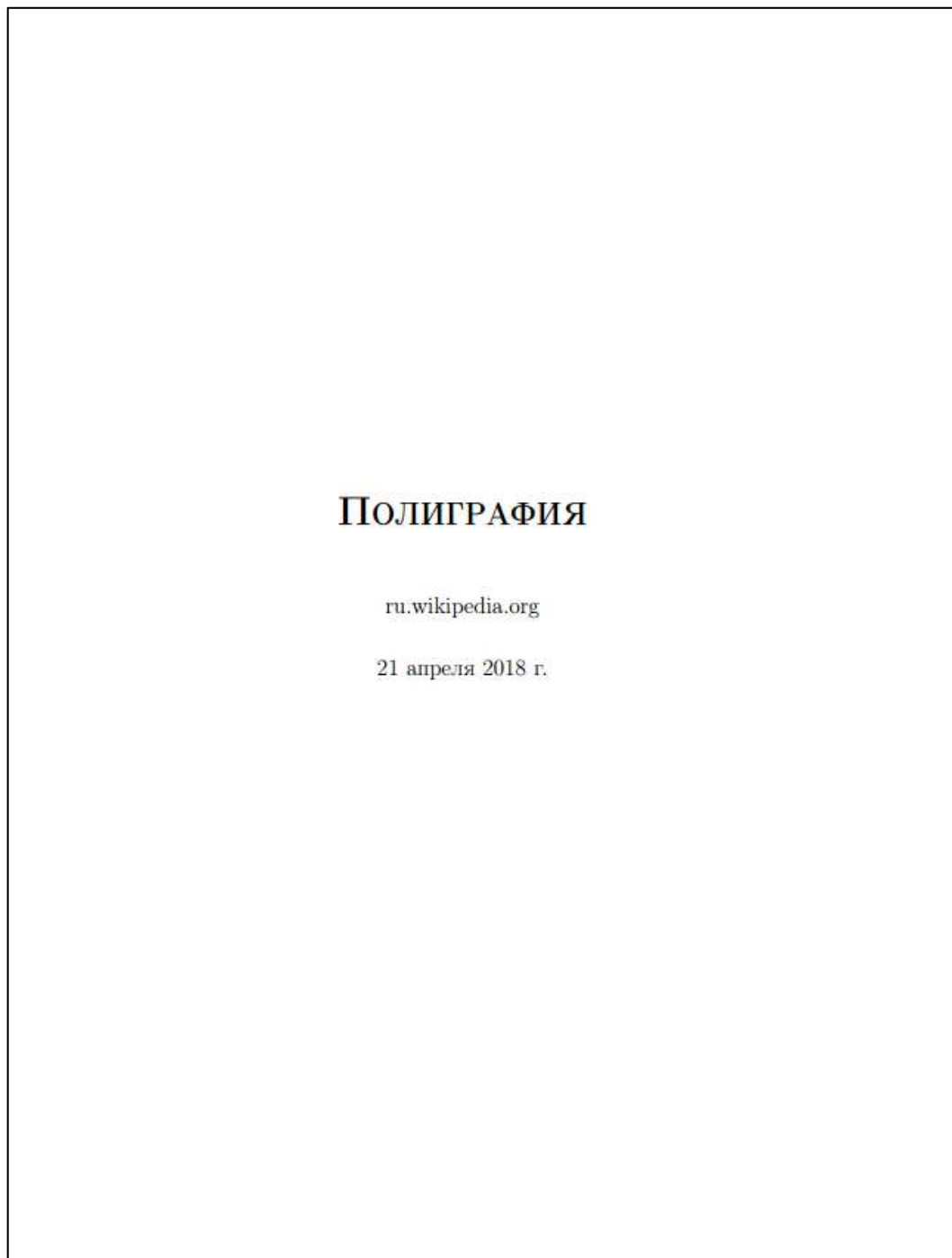


Рис. 3.10. Титул.

Оглавление

1 Основы полиграфии	5
Введение	5
1.1 Основные типы печати	5
1.2 Особые виды полиграфии	6
1.2.1 Ароматическая полиграфия	6
1.2.2 Стерео	6
1.3 Возможные дефекты	7
1.3.1 Отмарывание	7
1.3.2 Марашки	7
1.3.3 Разнооттеночность оттисков	7
1.3.4 Деформация растровых точек	7
1.3.5 Двоение печатных элементов	7
1.3.6 Муар	8
1.3.7 Полошение	8
1.4 Программы для автоматизации учета в типографиях	8
1.5 Перспективы полиграфической отрасли	8

Рис. 3.11. Оглавление.

Глава 1

Основы полиграфии

Введение

Полиграфия — отрасль промышленности, занимающаяся изготовлением печатной продукции, а именно книжно-журнальной, деловой, газетной, этикеточной и упаковочной продукции.

1.1 Основные типы печати

К основным типам печати относят: высокую печать, глубокую печать, плоскую печать, трафаретную печать.

Все способы печати являются разновидностями этих основных типов.

Выделяют следующие способы печати: офсетная, цифровая, цифровая офсетная, флексография, сублимация, шелкография, тиснение.

К плоской печати относят литографию, тампонную и ирисовую печать.

Рис. 3.12. Оформление введения и основных терминов.

окрашенными стёклами) стереопары или же при печати на лентиккулярном (линзовом) растре.

1.3 Возможные дефекты

1.3.1 Отмарывание

Отмарывание – переход краски с одного листа на оборот следующего.

Причины: превышение суммарного объема красок, неправильная сушка, быстрая отправка готового тиража в послепечатную обработку.

1.3.2 Марашки

Марашки – белые пятна, появляющиеся из-за отрыва в процессе печати частиц бумаги, которые потом прилипают к поверхности изделия, печатной форме или полотну.

1.3.3 Разнооттеночность оттисков

Разнооттеночность оттисков – не выдержан одинаковый цвет оттисков одного тиража.

1.3.4 Деформация растровых точек

Деформация растровых – дефект возникает в результате излишнего давления между цилиндрами или из-за разной скорости их работы.

1.3.5 Двоение печатных элементов

Двоение печатных элементов – появление второго, смещенного, изображения одного и того же элемента. Причины: недостаточное давление между цилиндрами; перекося бумаги.

Рис. 3.13. Оформление дополнительных терминов.

3.2 Перечисления

3.2.1 Оформление перечислений

Перечисления, или **списки** — это одна из наиболее удобных форм представления связанной информации.

Читателя может удивить, что эта тема не была обозначена ранее: ее логично было бы связать с понятием текстового абзаца. Однако мы решили несколько повременить с ней и рассмотреть ее на этом занятии.

LaTeX поддерживает три вида перечислений и каждое из них оформляется с помощью соответствующего окружения.

Нумерованный список

Окружение `enumerate`

Организует автоматически нумеруемый список.

Поддерживает четырехуровневое вложение.

Для указания одного пункта списка используется команда

`\item текст_пункта`

Допускается указать произвольный маркер:

`\item[маркер] текст_пункта`



Пример

```
\begin{enumerate}
  \item Пункт списка
  \item Пункт списка
  \item Пункт списка
\end{enumerate}
```


Результат

1. Пункт списка
2. Пункт списка
3. Пункт списка

Рис. 3.14. Оформление нумерованного списка.

Маркированный список

Окружение `itemize`

Организует маркированный список.

Поддерживает четырехуровневое вложение.

Для указания одного пункта списка используется команда

`\item текст_пункта`

Допускается указать произвольный маркер:

`\item[маркер] текст_пункта`



Пример

```
\begin{itemize}
  \item Пункт списка
  \item Пункт списка
  \item Пункт списка
\end{itemize}
```

Результат

- Пункт списка
- Пункт списка
- Пункт списка

Рис. 3.15. Оформление маркированного списка.

Список описаний

Окружение `description`

Организует список описаний (определений).

Для указания одного пункта списка используется команда

`\item[термин] текст_пункта`



Пример

```
\begin{description}
  \item[Термин 1] Пункт списка
  \item[Термин 2] Пункт списка
  \item[Термин 3] Пункт списка
\end{description}
```

Результат

Термин 1 Пункт списка

Термин 2 Пункт списка

Термин 3 Пункт списка

Рис. 3.16. Оформление списка описаний.

3.2.2 Пример оформления списков

Рассмотрим пример использования списков в одном документе.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}
```

```
\linespread{1.2}
```

```
\begin{document}
  Нумерованный список:
  \begin{enumerate}
    \item Пункт списка
    \item Пункт списка
    \item Пункт списка
  \end{enumerate}
```

```
  Маркированный список:
  \begin{itemize}
    \item Пункт списка
    \item Пункт списка
    \item Пункт списка
  \end{itemize}
```

```
  Список описаний:
  \begin{description}
    \item[Термин 1] Пункт списка
    \item[Термин 2] Пункт списка
```

```

\item[Термин 3] Пункт списка
\end{description}
\end{document}

```

Результат

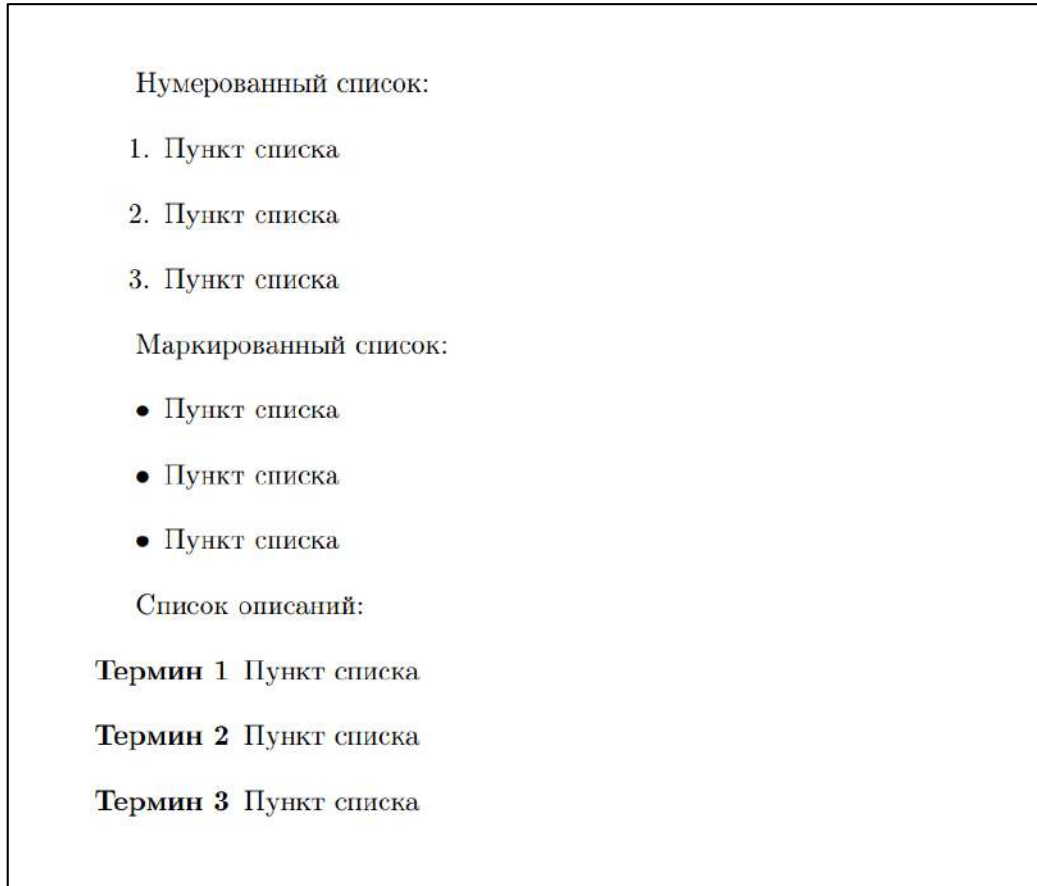


Рис. 3.17. Оформление всех видов списков в одном документе.

3.2.3 Настройка отступов списка

Из предыдущего примера можно заметить, что

- маркер пункта списка стоит левее, чем отступ красной строки и абзаца;
- между смежными пунктами списка интервал больше, чем интерлиньяж.

Для настройки подобных параметров может быть полезным пакет **enumitem**. Изучим некоторые его возможности на доработке предыдущего примера. Зададим следующие отступы:

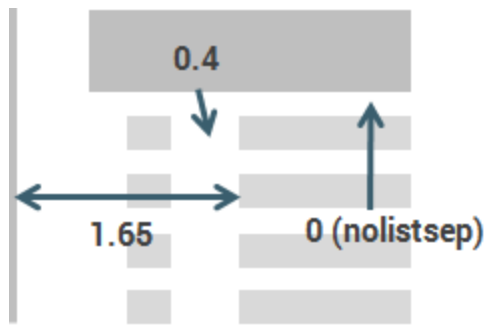


Рис. 3.18. Тонкая настройка оформления списков.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Пакет для настройки отступов у списков
\usepackage{enumitem}
\setlist{nolistsep, labelsep=0.4cm, leftmargin=1.65cm}

\linespread{1.2}

\begin{document}
  Нумерованный список:
  \begin{enumerate}
    \item Пункт списка
    \item Пункт списка
    \item Пункт списка
  \end{enumerate}

  Маркированный список:
  \begin{itemize}
    \item Пункт списка
    \item Пункт списка
    \item Пункт списка
  \end{itemize}

  Список описаний:
  \begin{description}
    \item[Термин 1] Пункт списка
    \item[Термин 2] Пункт списка
    \item[Термин 3] Пункт списка
  \end{description}
\end{document}
```

Результат

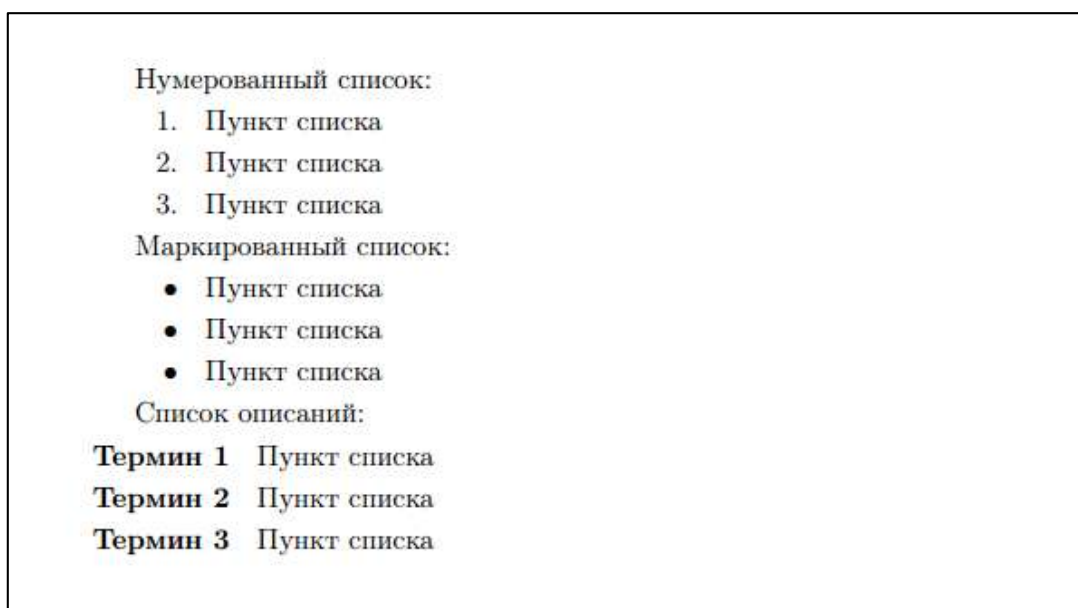


Рис. 3.19. Изменение стандартных параметров форматирования отступов у списков.

Вопросы для самопроверки

1. Перечислите виды списков, поддерживаемых LaTeX.
2. Почему списки рекомендуется оформлять, используя встроенные окружения?
3. Как дополнительно настроить отступы в списках?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Закрепление пройденного материала

1. Проверьте работу всех трех видов списков из текущего занятия.
2. Изучите работу с дополнительными параметрами форматирования элементов списка.

2. Вложенные списки

Реализуйте сложный список, вкладывая списки в соответствующие пункты (см. рис. 3.20).

При необходимости, настройте дополнительные интервалы для элементов списка.

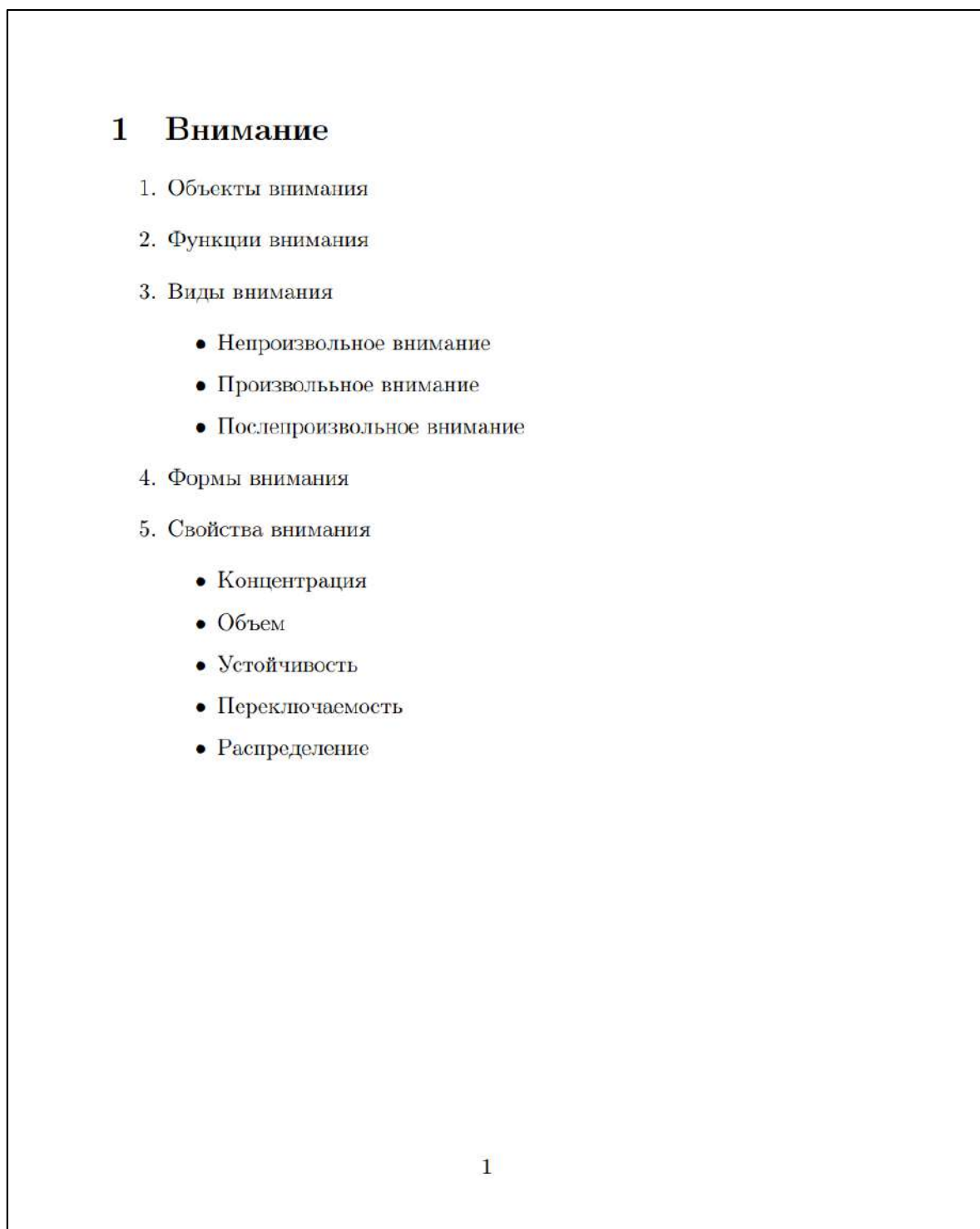


Рис. 3.20. Требуемый результат.

3.3 Изображения

3.3.1 Особенности работы с изображениями

Пакет **graphicx** и драйверы

Наиболее известным пакетом для работы с изображениями в LaTeX является пакет **graphicx**. Он позволяет одновременно работать с векторной и растровой графикой. В опциях можно указать используемый драйвер:

```
\usepackage[драйвер]{graphicx}
```

- **dvips** (по умолчанию) используется при компиляции LaTeX для получения DVI-файла;
- **dvipdfm**, если документ компилируется LaTeX, и полученный DVI-файл предполагается конвертировать в PDF при помощи dvipdfm;
- **pdftex** (по умолчанию при компиляции с помощью pdflatex), если документ предполагается компилировать pdftex, чтобы непосредственно получить PDF-файл.

В текущем курсе нам достаточным будет использование возможностей pdflatex, поэтому в преамбуле указываем команду

```
\usepackage{graphicx}
```

LaTeX способен собирать изображения, находящиеся в разных каталогах относительно основного файла. Рационально заранее сохранить их в отдельный каталог(и) и указать к ним путь командой **\graphicspath**:

```
% в одном каталоге
\graphicspath{{pictures/}}
% в каталогах
\graphicspath{{pictures/}{logo/}{photos/my/} }
```

Дополнительно можно указать, какие из типов файлов можно считать графическими:

```
\DeclareGraphicsExtensions{.pdf,.png,.jpg,.bmp}
```

Эта команда позволит указывать только имя подключаемого изображения, без явного указания расширения (см. далее).

Формат EPS

Encapsulated PostScript (EPS) – формат графических файлов, часто используемый в полиграфии. Этот формат может содержать растровые и векторные изображения, текст, а также их комбинации.

Исторически EPS является наиболее используемым форматом изображений в LaTeX. Это обосновывается, прежде всего, возможностью его масштабирования.

Но современные условия потребовали работы и с другими форматами, например, PNG, PDF, JPG и т.д. «Чистый» LaTeX не предполагает работы с этими форматами, либо имеет определенную специфику: чтобы вставить растровое изображение, LaTeX должен знать точные размеры бокса. А вот расширения, типа PDFLaTeX работают с изображениями достаточно успешно.

Проект Overleaf.com поддерживает необходимые возможности работы с PDFLaTeX. Дополнительных настроек не требуется!

3.3.2 Работа с изображениями

Вставка изображений

```
\includegraphics{название_изображения}
```

Вставляет изображение.

Пример

```
\includegraphics{kotiki}  
% или  
\includegraphics{kotiki.png}
```

По умолчанию LaTeX в первой команде будет искать файл «kotiki.eps». Но если указать команду `\DeclareGraphicsExtensions`, то ищутся изображения с таким же именем уже среди всех перечисленных форматов. Вторая команда определенно указывает на файл и расширение.

Рассмотрим пример вставки изображения без какой-либо предварительной настройки его размеров.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Подключаем пакет для работы с изображениями
\usepackage{graphicx}

% Определяем расширения файлов с изображениями,
% для которых установлен короткий доступ
\DeclareGraphicsExtensions{.pdf,.png,.jpg}

\linespread{1.15}

\begin{document}

    \section{Вставка изображения}

    \includegraphics{kotiki}

\end{document}
```

Результат

1 Вставка изображения



1

Рис. 3.21. Вставка изображения.

Предыдущий пример четко показывает, что по умолчанию изображение вставляется в реальном размере. Нужна его настройка.

Настройка изображений

```
\includegraphics[опции]{название_изображения}
```

Вставляет изображение и форматирует его по указанным опциям:

- ***scale=число*** – указывает масштаб изображения относительно истинных размеров ($1.0 = 100\%$);
- ***width=ширина*** – ширина, под которую подгоняется изображение;
- ***height=высота*** – высота, под которую подгоняется изображение;
- ***angle=число*** – угол поворота изображения.

В следующем примере демонстрируется вставка одного и того же изображения с разным масштабом. Пустая строка между командами равносильна новому абзацу, поэтому изображения находятся в отдельных боксах (рис. 3.22).

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{graphicx}
\DeclareGraphicsExtensions{.pdf,.png,.jpg}

\linespread{1.15}

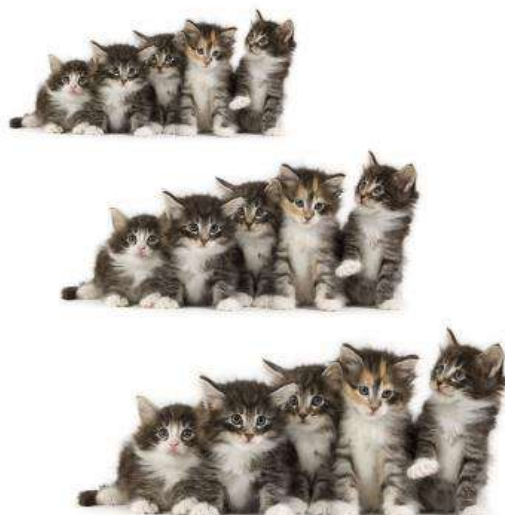
\begin{document}
  \section{Вставка изображения}
  \includegraphics[scale=0.2]{kotiki}   % шириной в 20%

  \includegraphics[scale=0.25]{kotiki} % шириной в 25%

  \includegraphics[scale=0.3]{kotiki}  % шириной в 30%
\end{document}
```

Результат

1 Вставка изображения



1

Рис. 3.22. Масштабирование изображений.

Масштабирование изображений относительно их исходного размера предполагает знание их точного размера: в противном случае нет гарантии, что изображение даже после изменения масштаба поместится на листе, либо окажется слишком маленьким. Поэтому опцию `scale` обычно используют с такими параметрами как `\linewidth` (ширина строки), `\textwidth` (ширина текста на странице), `\textheight` (высота текста на странице) и др. Они подгоняют масштаб изображения относительно указанных параметров, что, в частности, удобно при вставке нескольких изображений в ряд (рис. 3.23).

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Подключаем пакет для работы с изображениями
\usepackage{graphicx}

% Определяем расширения файлов с изображениями,
% для которых установлен короткий доступ
\DeclareGraphicsExtensions{.pdf,.png,.jpg}

\linespread{1.15}

\begin{document}

    \section{Вставка изображения}

    % 100% по ширине текста
    \includegraphics[width=1.0\textwidth]{kotiki}

    \includegraphics[width=0.5\textwidth]{kotiki}
    % два одинаковых изображения в один ряд
    \includegraphics[width=0.5\textwidth]{kotiki}

\end{document}
```

Результат

1 Вставка изображения



1

Рис. 3.23. Масштабирование изображений относительно ширины области печати.

Позиционирование и описание изображений

Окружение figure

Позволяет осуществить более полную настройку изображения: определить его положение, назначить подпись к изображению и метку перекрестной ссылки.

LaTeX будет стараться найти наиболее оптимальное положение изображения, что может не совпадать с положением команд в разметке. В этом случае используется опция:

- ***h*** – поместить изображение «здесь» (однако в зависимости от свободного пространства окончательное решение принимает LaTeX);
- ***p*** – поместить изображение на отдельной странице;
- ***t*** – поместить изображение сверху;
- ***b*** – поместить изображение снизу;
- ***!*** – поместить изображение именно так, как хочет пользователь (т.е. «желания» LaTeX игнорируются).

Пример

```
% Желательно поместить здесь
\begin{figure}[h]
. . .
\end{figure}

% Поместить здесь и точка!
\begin{figure}[h!]
. . .
\end{figure}

% Поместить на отдельной странице и только сверху!
\begin{figure}[pt!]
. . .
\end{figure}
```


Ниже приведен пример качественно оформленного изображения (внешний вид документа см. на рис. 3.24).

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Подключаем пакет для работы с изображениями
\usepackage{graphicx}

% Определяем расширения файлов с изображениями,
% для которых установлен короткий доступ
\DeclareGraphicsExtensions{.pdf,.png,.jpg}

\linespread{1.15}

\begin{document}

    \section{Вставка изображения}

    \begin{figure}[h!]
        \centering          % расположить по центру
        \includegraphics[width=0.8\textwidth]{kotiki}
        \caption{А это наши котики!    % подпись
        \label{fig:kotiki}    % метка для ссылки
    \end{figure}

\end{document}
```

Результат

1 Вставка изображения



Рис. 1: А это наши котики!

1

Рис. 3.24. Форматирование изображения с подключением подписи и метки для ссылки.

Вопросы для самопроверки

1. Какие драйверы для работы с изображениями поддерживает LaTeX?
2. Перечислите особенности формата EPS.
3. Какие опции изображения можно настроить?
4. Почему масштабирование изображения относительно его реального размера нерационально?
5. Укажите основные возможности окружения figure.

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Вставка изображений в текст документа

Изучите примеры работы с изображениями в текущем занятии (можете взять любое изображение).

Создайте документ (рис. 3.25-рис. 3.26) по следующему шаблону:

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage[
  left=3cm,
  right=2cm,
  top=2cm,
  bottom=2cm
]{geometry}

\usepackage{graphicx}
\DeclareGraphicsExtensions{.pdf,.png,.jpg}
```

```

% Настройка списков
\usepackage{enumitem}
\setlist{nolistsep, labelsep=0.4cm, leftmargin=1.85cm}

% Красная строка у первого абзаца после заголовка
% (по умолчанию не ставится)
\usepackage{indentfirst}

% Здесь укажите свойства
\linespread{1.15}
\parindent = 0.75cm

\begin{document}

    % Текст документа

\end{document}

```

Если вы работаете с Overleaf, то предварительно изображения нужно загрузить в каталог проекта!

2. Изображения на отдельном листе

Повторите основные приемы позиционирования изображений.

На отдельном листе оформить следующее ниже изображение (рис. 3.27). Предполагается, что оно всегда остается только на одном листе. Можно взять любое изображение.

Результат 1

1 Фракталы

Фрактал (лат. fractus — дроблёный, сломанный, разбитый) — множество, обладающее свойством самоподобия (объект, в точности или приближённо совпадающий с частью себя самого, то есть целое имеет ту же форму, что и одна или более частей).

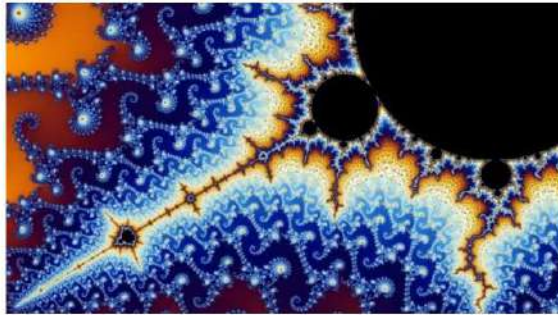


Рис. 1: Фрактал Мандельброта.

1.1 Фракталы в природе

В живой природе:

- кораллы;
- морские звезды, ежи, раковины;
- цветы и растения (брокколи, капуста);
- кроны деревьев и листья растений (см. рис. 3);
- кровеносная система и бронхи людей и животных.



Рис. 2: Фрактальная структура растений.

В неживой природе:

- границы географических объектов (стран, областей, городов);
- береговые линии;
- горные хребты;
- снежинки;
- облака;
- кристаллы (например, кристалл висмута, рис. 3).

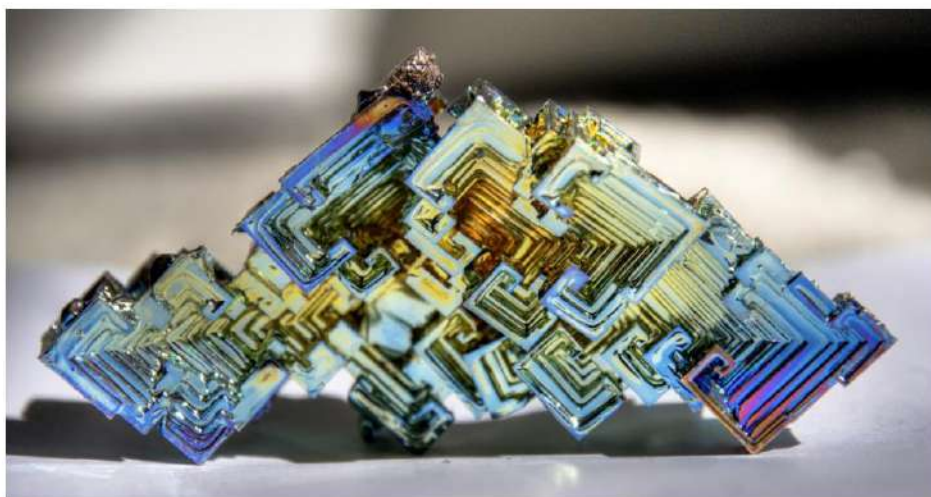
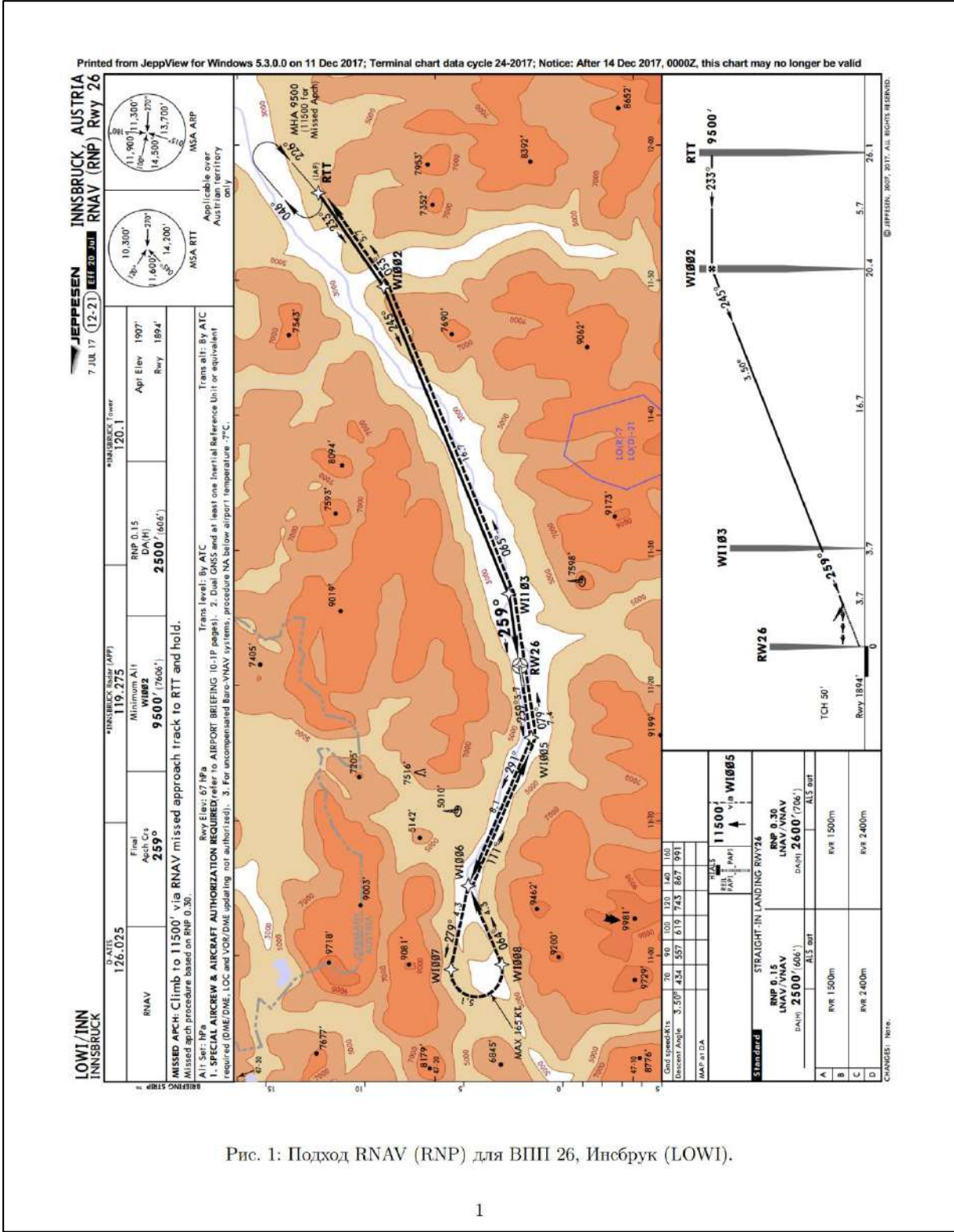


Рис. 3: Кристалл висмута.

Результат 2



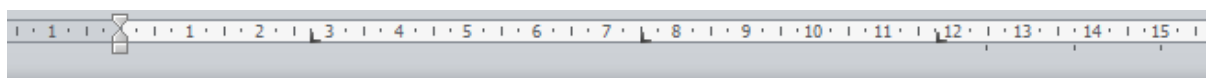
3.4 Таблицы

Таблица – это одна из наиболее простых конструкций для оформления структурированных данных. Текст, оформленный с помощью таблицы, существенно проще воспринимать и анализировать. И зачастую содержимое таблиц описано более лаконично, избегаются повторы текстовой информации.

За оформление таблиц и похожих структур в LaTeX отвечают два стандартных окружения: **tabbing** и **tabular**.

3.4.1 Окружение **tabbing**

Указанное окружение предназначено для табулирования текстовой информации. Этот прием хорошо известен, например, активным пользователям MS Word: с помощью специальных символов табуляции можно устанавливать соответствующее положение курсора в строке или выравнивать содержимое нескольких строк по указанному отступу (рис. 3.28).



№	ФИО	Возраст	Стаж (лет)
1	Иванов П.М.	45	17
2	Петров Д.А.	56	28
3	Егоров	48	21

Рис. 3.28. Табулирование текста в MS Word (позиции табуляторов отмечены на линейке).

Именно поэтому табулятор нельзя назвать инструментом для полноценной верстки таблицы: он лишь позволяет имитировать табличное оформление.

Окружение **tabbing** поддерживает следующие команды:

Команда	Описание
\=	Задаёт позицию табулятора (курсора). Дополнительно можно указать пробел: \hspace{x}\=
\kill	Убирает строку. Как правило, это первая (форматирующая) строка, которая указывает шаблон отступов.
\>	Сдвигает курсор к следующей позиции.
\\	Начало новой строки.
*	Запрет разрыва строки.
\'	Выравнивание вправо (относительно табулятора).
\+	Сдвигает все позиции вправо.
\-	Сдвигает все позиции влево.

Пример

```

\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}

  \begin{tabbing}
    МММ \= ММММММММММ \= МММММ \= МММММММ \kill
    № \> ФИО \> Возраст \> Стаж (лет)\\
    1 \> Иванов П.М. \> 45 \> 17\\
    2 \> Петров Д.А. \> 56 \> 28\\
    3 \> Егоров К.А. \> 48 \> 21
  \end{tabbing}

  \begin{tabbing}
    \hspace{3em}\= \hspace{12em}\= \hspace{6em}\=
    \hspace{6em}\kill
    № \> ФИО \> Возраст \> Стаж (лет)\\
    1 \> Иванов П.М. \> 45 \> 17\\
    2 \> Петров Д.А. \> 56 \> 28\\
    3 \> Егоров К.А. \> 48 \> 21
  \end{tabbing}

\end{document}

```

Результат

№	ФИО	Возраст	Стаж (лет)
1	Иванов П.М.	45	17
2	Петров Д.А.	56	28
3	Егоров К.А.	48	21

№	ФИО	Возраст	Стаж (лет)
1	Иванов П.М.	45	17
2	Петров Д.А.	56	28
3	Егоров К.А.	48	21

Рис. 3.29. Пример оформления текста с помощью окружения `tabbing`.

В первом примере шаблон задается с помощью устранимой строки, где буква `M` резервирует позицию одного символа (можно было указать любой другой символ). Во втором примере расстояние между табуляторами устанавливается горизонтальным пробелом.

Перечислим особенности и ограничения окружения `tabbing`:

1. Позиции всех табуляторов и отступов должны быть заданы пользователем явно, иначе текст соседних табуляторов может перекрывать его.
2. Табуляторы работают только для текстовой разметки.
3. Разделительных линий (границ) `tabbing` не поддерживает.
4. Вложение также не поддерживается.
5. При необходимости содержимое «колонок» переносится по строкам на следующую страницу.

3.4.2 Окружение `tabular`

Для отрисовки таблицы с границами используется окружение **`tabular`**. Основным аргументом окружения является команда указания колонки, которая также определяет и способ выравнивания: **`l`** (по левому краю), **`c`** (по центру), **`r`** (по правому краю). Вертикальную линию позволяет рисовать команда `|`. Например,

```
\begin{tabular}{l|r|cc}
```

отрисовывает таблицу в четыре колонки, первая колонка выравнивается по левому краю, вторая – по правому, третья и четвертая – по центру; вертикальные границы есть только у второй колонки.

Содержимое таблицы формируется построчно. Содержимое ячеек в строке разделяется символом **&**, а каждая строка завершается командой разрыва строки ****. Горизонтальную линию позволяет сделать команда **\hline**.

Автоматически масштабируемые таблицы

По умолчанию LaTeX подгоняет ширину колонок по содержанию ячеек.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}

  \begin{tabular}{l||l||cc}
    \hline
    № & ФИО & Возраст & Стаж (лет)\\
    \hline\hline
    1 & Иванов П.М. & 45 & 17 \\
    2 & Петров Д.А. & 56 & 28 \\
    3 & Егоров К.А. & 48 & 21
  \end{tabular}

\end{document}
```

Результат

№	ФИО	Возраст	Стаж (лет)
1	Иванов П.М.	45	17
2	Петров Д.А.	56	28
3	Егоров К.А.	48	21

Рис. 3.30. Пример верстки автоматически масштабируемой таблицы.

Ячейки можно оставлять пустыми.

Пример

```
\begin{tabular}{l|l|cc}  
 \hline  
   & ФИО & Возраст & Стаж (лет) \\ \hline \hline  
 1 & Иванов П.М. & & 17 \\ \hline  
 2 & Петров Д.А. & 56 & 28 \\ \hline  
 3 & Егоров К.А. & & \\ \hline  
\end{tabular}
```

Результат

	ФИО	Возраст	Стаж (лет)
1	Иванов П.М.		17
2	Петров Д.А.	56	28
3	Егоров К.А.		

Рис. 3.31. Пример верстки автоматически масштабируемой таблицы (с пустыми ячейками).

Таблицы с фиксированной шириной колонок

Если ширина колонок должна быть задана точно, используется команда

```
\begin{tabular}{p ширина}
```

которую требуется указать для каждой колонки. Команды выравнивания при этом не работают: оно осуществляется по ширине.

Пример

```
\begin{tabular}{p{1cm}||p{5cm}||p{2.5cm}p{2.5cm}}
\hline
№ & ФИО & Возраст & Стаж (лет)\\
\hline\hline
1 & Иванов П.М. & 45 & 17 \\
2 & Петров Д.А. & 56 & 28 \\
3 & Егоров К.А. & 48 & 21
\end{tabular}
```

Результат

№	ФИО	Возраст	Стаж (лет)
1	Иванов П.М.	45	17
2	Петров Д.А.	56	28
3	Егоров К.А.	48	21

Рис. 3.32. Пример таблицы с определенно заданной шириной колонок.

Особенности окружения `tabular`

1. Колонки растягиваются пропорционально содержимому.
2. Допускается горизонтальное выравнивание для каждой колонки.
3. Таблицы могут быть вложены.
4. Таблицы не переносятся частями на другую страницу: для этого потребуется подключение дополнительных пакетов.

Разумеется – это далеко не все возможности `LaTeX` в вопросе верстки таблиц. Допускается объединение ячеек, настройка их фона и других свойств. Делается это, в том числе, с помощью дополнительных пакетов.

Вопросы для самопроверки

1. Чем отличаются таблицы от табулируемого текста?
2. Опишите механизм использования табуляторов.
3. Напишите фрагмент кода разметки, рисующий таблицу на 8 строк и 8 колонок, для которой присутствует внешняя граница, текст в первых четырех колонках выравнивается по левому краю, а в остальных – по центру.
4. Как создать таблицу с пустыми ячейками?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Выравнивание текста с помощью табуляторов

С помощью табуляторов оформите текст первого пункта и треугольник Паскаля, как показано на рис. 3.33.

2. Оформление таблиц

Постройте таблицу умножения простых чисел (рис. 3.34). Установите рубленый шрифт, полужирное начертание элементов на диагонали. Таблицу выровнять по центру.

Оформите таблицу, изображенную на рис. 3.35.

1 Обычное табулирование

№	ФИО	Группа	Участие
1	Петрова Е.Н.	ИМ-118	+
2	Рыжова П.Е.	ФМ-119	+
3	Ершов А.М.	ИИ-119	–
4	Авдеев М.Р.	ИИ-119	+

2 Треугольник Паскаля

1						
1	1					
1	2	1				
1	3	3	1			
1	4	6	4	1		
1	5	10	10	5	1	
1	6	15	20	15	6	1

Рис. 3.33. Требуемый результат для задания 1.

Таблица умножения простых чисел

	2	3	5	7	11	13
2	4	6	10	14	22	26
3	6	9	15	21	33	39
5	10	15	25	35	55	65
7	14	21	35	49	77	91
11	22	33	55	77	121	143
13	26	39	65	91	143	169

Рис. 3.34. Требуемый результат для задания 2 (таблица №1).

Период	Концепция использования информации	Вид	Цель использования
1950 – 1960 гг.	Бумажный поток	ИС обработки расчетных документов на электронных бухгалтерских машинах	Повышение скорости. Упрощение процедуры обработки счетов
1960 – 1970 гг.	Подготовка отчетов	Управленческие ИС для производственной информации	Ускорение процесса отчетности
1970 – 1980 гг.	Управленческий контроль реализации	Системы поддержки принятия решений. Системы для высшего звена управления	Выработка наиболее рационального решения
с 1980 гг.	Информация – стратегический ресурс конкуренции	Стратегические ИС. Автоматизированные офисы	Выживание и процветание фирмы

Рис. 3.35. Требуемый результат для задания 2 (таблица №2).

3.5 Библиография и перекрестные ссылки

3.5.1 Библиография

LaTeX поддерживает разные подходы к генерации библиографических списков. К стандартному и, пожалуй, наиболее простому относят возможности окружения **thebibliography**.

Более развитым подходом является использование библиографических баз. Здесь каждый источник описывается по определенным ключам, что улучшает структуризацию списка и также полезно для поисковых систем, которые способны обрабатывать данные таких баз. Ярким примером является ViBTeX.

3.5.2 Окружение thebibliography

Окружение thebibliography

Организует автоматически нумеруемый список библиографических источников. Также печатает заголовок «Список литературы».

Для указания источника используется команда

`\bibitem{название} описание_источника`

Для ссылки на источник в тексте используется команда

`\cite{название}`

У окружения имеется обязательный и «необычный» параметр — это последовательность символов, число которых равно количеству разрядов в наибольшем номере библиографии. Например, указав запись

`\begin{thebibliography}{99}`

на номер отводится пространство в два разряда. (Само число 99 ни на что не влияет)

По умолчанию заголовок библиографии не вносится в оглавление. Однако это легко исправить командой `\addcontentsline` (см. раздел 0).

Работу окружения `thebibliography` проясняет следующий модельный пример:

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\linespread{1.3}

\begin{document}
  \tableofcontents

  \vspace{10em}

  . . .

  Подробнее работа с коллекциями рассматривается
  в работе~\cite{Richter_1}.

  . . .

  % Добавить в оглавление
  \addcontentsline{toc}{section}{Список литературы}

  \begin{thebibliography}{99}
    \bibitem{McConnel} Макконел С. Совершенный код.
      Мастер-класс / Пер. с англ. --- М.:
      Издательство <<Русская редакция>>б 2010.
    ---
      889 стр.: ил.

    \bibitem{Richter_1} Рихтер Дж. Программирование
      на платформе .NET Framework 4.5 на языке
      C\#. 4-е изд. --- СПб.: Питерб 2013. ---
      896 стр.: ил. -- (Серия <<Мастер-
      класс>>).
  \end{thebibliography}

\end{document}
```

Результат

Содержание	
Список литературы	1
...	
Подробнее работа с коллекциями рассматривается в работе [2].	
...	
Список литературы	
[1] Макконел С. Совершенный код. Мастер-класс / Пер. с англ. — М.: Издательство «Русская редакция»б 2010. — 889 стр.: ил.	
[2] Рихтер Дж. Программирование на платформе .NET Framework 4.5 на языке C#. 4-е изд. — СПб.: Питерб 2013. — 896 стр.: ил. — (Серия «Мастер-класс»).	
1	

Рис. 3.36. Пример генерации списка литературы и перекрестных ссылок на источники.

Необходимо отметить, что при перестановке источников местами все ссылки автоматически обновляются. Если команда `\cite` ссылается на несуществующий источник, то он помечается знаком вопроса.

3.5.3 Ссылки

Перекрестные ссылки

Базовый механизм перекрестных ссылок LaTeX строится на создаваемых пользователем метках.

```
\label{метка}
```

Команда определяет именованную метку, которая связывается с объектом.

Как правило, метки назначаются заголовкам, таблицам, изображениям, схемам, формулам, теоремам и т.д. Т.е. метка может быть привязана к любому объекту!

Для обращения к метке доступны две команды:

- `\ref{метка}` – вставляет номер объекта, привязанного к указанной метке;
- `\pageref{метка}` – вставляет номер страницы, на которой расположен объект.

Соглашение о названии меток

Пользователь вправе использовать любое название метки. Главное, чтобы оно не содержало пробелов или других спецсимволов и не совпадало с ключевыми командами.

Однако в большом документе число перекрестных ссылок может быть столь велико, что их грамотное именование становится неприятной проблемой. Именно поэтому хорошей практикой является использование некоторого соглашения по именованию меток. Рациональнее всего разделить метки по категориям объектов, к которым они привязываются.

Пользователи LaTeX обычно используют префиксы в названии меток, например:

`\label{sec:intro}`

как следует ожидать, привязана к секции (заголовку) введения, а

`\label{def:psychology}`

вероятно, к определению термина «психология».

В следующей таблице приведены некоторые общеупотребительные префиксы:

Префикс	Блок (англ.)	Блок (рус.)
ch:	chapter	глава
sec:	section	секция
subsec:	subsection	подсекция
fig:	figure	фигура, изображение
tab:	table	таблица
eq:	equation	формула
lst:	code listing	листинг кода
itm:	enumerate list item	пункт нумерованного списка
alg:	algorithm	алгоритм
app:	appendix	приложение
thrm:	theorem	теорема

Придумывайте меткам осмысленные и лаконичные имена. Избегайте чисел в названии меток.

В следующем примере продемонстрировано использование меток для разных объектов:

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Подключаем пакет для работы с изображениями
\usepackage{graphicx}

\begin{document}
  \large

  \section{Введение}\label{sec:intro}

  \section{Перекрестные ссылки}
  Согласно разделу~\ref{sec:intro}, ...

  Подробно мы уже рассматривали этот вопрос на
  стр.~\pageref{sec:intro} ...

  На рис.~\ref{fig:kotiki} вы можете видеть наших
  замечательных котиков:

  \begin{figure}[th]
    \centering

    \includegraphics[width=1.0\linewidth]{kotiki.png}\\
    \caption{А это наши котики.}
    \label{fig:kotiki}
  \end{figure}

\end{document}
```

Результат

1 Введение

2 Перекрестные ссылки

Согласно разделу 1, ...

Подробно мы уже рассматривали этот вопрос на стр. 1 ...

На рис. 1 вы можете видеть наших замечательных котиков:



Рис. 1: А это наши котики.

Рис. 3.37. Пример работы с перекрестными ссылками на изображения.

Гиперссылки

Наиболее мощным пакетом для работы с гиперссылками является пакет **hyperref**. Он позволяет создавать как ссылки на внешние ресурсы, так и на объекты внутри документа. Также этот пакет позволит настроить навигацию документа и внедрить ее в PDF-файл. Рассмотрим некоторые возможности этого пакета [37,38].

Для указания ссылок на внешние ресурсы используется команда **\href**:

```
\href{ссылка}{текст_ссылки}
```

Ссылку можно оформить моноширинным шрифтом (т.е. ширина всех символов одинакова), указав ее в команде **\url**:

```
\url{ссылка}
```

Организовать ссылку на формулу, раздел и любой другой объект с меткой позволяет команда **\hyperref**:

```
\hyperref[метка]{текст ссылки}
```

(это более развитый вариант команды **\ref** с той лишь разницей, что текст ссылки может быть любым).

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}
```

```
\usepackage{hyperref}
```

```
\begin{document}
```

Ссылка на сайт

```
\href{https://www.google.ru}{google.com}.
```

Просто ссылка моноширинным шрифтом:

```
\url{https://www.google.ru}.
```

Согласно `\hyperref[formula]{формуле \ref*{formula} }`

```
\begin{equation}\label{formula}
\sin^2 x + \cos^2 x = 1.
\end{equation}
```


\end{document}

Результат

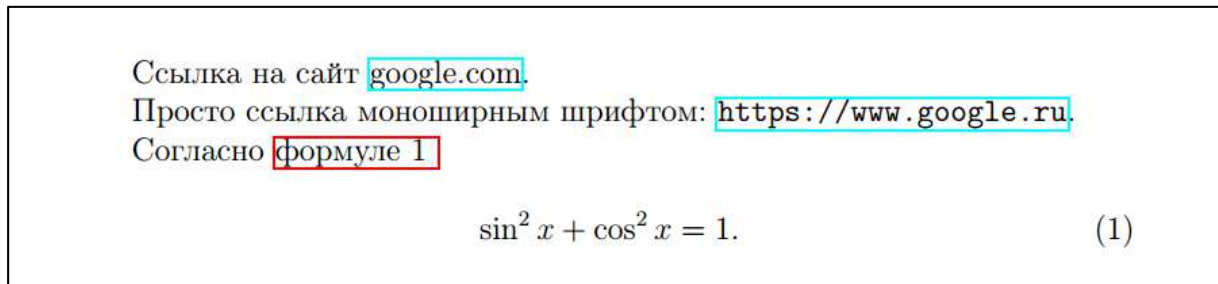


Рис. 3.38. Пример использования разных типов гиперссылок.

Как следует из примера, оформление ссылок по умолчанию мало привлекательно (заметим, что рамочка вокруг текста ссылки отображается лишь при отображении документа на экране, но не печати). Однако их можно настроить с помощью команды **\hypersetup**. В параметре этой команды указывают переменные, настраивающие цвета ссылок и их связь с PDF-документом; параметров команды достаточно много и при желании читатель без труда найдет полное описание. Мы ограничимся лишь небольшим примером и несколькими переменными:

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{xcolor}      % Пакет для работы с цветами
\usepackage{hyperref}    % Пакет для работы с ссылками

% Настройка нек. параметров ссылок
\hypersetup{
  colorlinks = true,      % Ссылки цветные? - Да
  linkcolor = {red!75!black}, % Внутренние ссылки -
                             % Красно-черный (75%)
  urlcolor = blue,        % Цвет ссылок на внешние
                             % (Интернет) ресурсы
  bookmarks = true,       % Создавать закладки-
                             % оглавление в PDF? - Да
}
```

```

\begin{document}

    Ссылка на сайт
    \href{https://www.google.ru}{google.com}.

    Просто ссылка моноширинным шрифтом:
    \url{https://www.google.ru}.

    Согласно \hyperref[formula]{формуле \ref*{formula} }

    \begin{equation}\label{formula}
        \sin^2 x + \cos^2 x = 1.
    \end{equation}

\end{document}

```

Результат

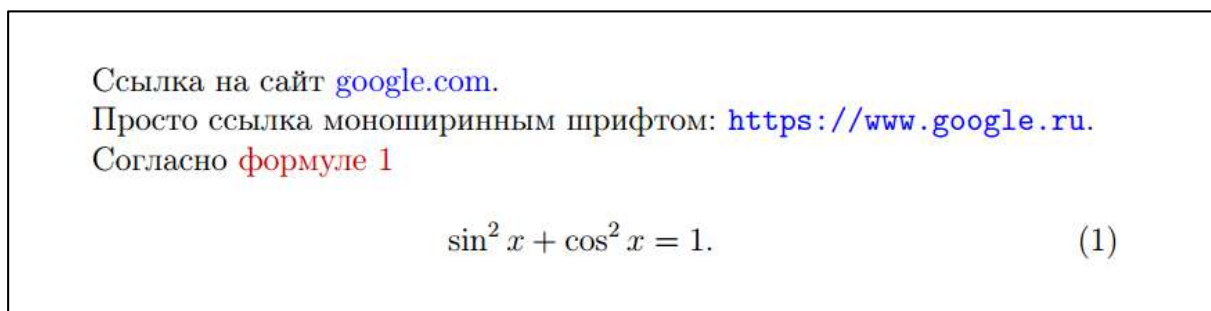


Рис. 3.39. Изменение стандартного форматирования гиперссылок: рамки убраны, ссылки на формулы отмечаются красным цветом.

Вопросы для самопроверки

1. Как создать библиографический список в LaTeX?
2. Что такое библиографическая база и в чем ее преимущества?
3. Чем полезны перекрестные ссылки?
4. Почему рекомендуется систематизировать названия меток?
5. Как указать ссылку на внешний ресурс?
6. Что делает команда `\hypersetup`?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Оформление библиографии

Скопируйте фрагмент следующего документа:

```
\documentclass[12pt]{book}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}
```

```
% Здесь укажите свойства
\linespread{1.45}
\parindent = 1.25cm
```

```
\begin{document}
  \chapter{Издательская система \LaTeX}
  \addcontentsline{toc}{section}{Введение}
  \section*{Введение}
  Исторически книга (печатное издание) является
  дефакто стандартом по хранению и распространению
  текстовой информации. И даже на современном этапе
  с развитием ИКТ и мультимедийных технологий
  представление информации в форме книжного листа
  является наиболее распространенным. Многочисленные
  исследования психологов показали влияние качества
  и формы представления тексто-графической
  информации на процесс восприятия и
  понимания. [1], [2]
```

В подавляющем большинстве учебных организаций используется пакет MS Office или его аналоги. Выбор оправдан простотой и реализацией большинства часто востребованных функций редактора. Однако, во-первых, он проприетарный, во-вторых, зачастую требует от автора излишней концентрации на форматировании документа, чем на его логической структуре и смысловом наполнении. Между тем, в научной отрасли давно себя зарекомендовала

настольная издательская система LaTeX, способная автоматизировать подготовку качественно оформленного электронного и бумажного документа, что позволяет автору сконцентрироваться на содержании своей работы и контролировать весь процесс его разработки.

LaTeX – это настольная издательская система, ориентированная на набор и верстку научных текстов, в частности, содержащих математические формулы, диаграммы, графики и т.д., является общепринятым форматом для научных публикаций. LaTeX находит широкое применение и в практике педагогов зарубежных школ, колледжей. В Р.Ф. LaTeX на этой нише мало популярен.

LaTeX упрощает верстку документов со стандартным оформлением (лекции, статьи, книги, учебно-методические пособия, эссе, рефераты, курсовые и дипломные работы, диссертации, др.). Пользователи LaTeX отмечают преимущество системы в вопросах разработки больших документов. Все вышесказанное подтверждает актуальность использования издательской системы LaTeX. [3], [4], [5].

.

1. Валгина, Н.С. Теория текста. Учебное пособие – М.: Логос, 2003. – 173 с.
2. Ширинкина, Л.В. Методы восприятия текста / Актуальные проблемы философии, социологии и политологии, экономики и психологии: 22 Материалы юбилейной научной студенческо-аспирантской конференции (апрель 2001)/ Перм.ун-т. – Пермь, 2001. – Вып.4 – с. 138-141.
3. Львовский, С.М. Набор и верстка в системе LaTeX / С.М. Львовский. – М.: МЦНМО, 2014. – 398 с.
4. Львовский, С.М. Работа в системе LaTeX / Львовский С.М. – М. Национальный Открытый Университет «ИНТУИТ», 2007. – 465 с.

5. Котельников, И. А., Чеботаев, П. З. LaTeX2e по-русски. – Новосибирск: Сибирский Хронограф, 2004. – 496 с.
`\end{document}`

Оформите в нем библиографию согласно правилам LaTeX (источники указаны в коде разметки).

В тексте исправьте ссылки на каждый источник, используя команды перекрестных ссылок `\cite` (сейчас они набраны вручную).

Если задание выполнено верно, то перекрестные ссылки будут указывать на соответствующие источники, а список литературы примет следующий вид (рис. 3.40).

2. Оформление гиперссылок

Скопируйте следующий код:

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage[
    top=2.5cm,
    bottom=2cm,
    left=3cm,
    right=1.5cm
]{geometry}

\begin{document}
  \begin{center}
    \Large\textbf{Электронные ресурсы по языку\backslash
    программирования C\#}
  \end{center}

  \begin{enumerate}
    \item Руководство по программированию на
      C\# от MSDN:
      https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/.

    \item Полное руководство по языку
      программирования C\# 7.0 и платформе
      .NET 4.7:
      https://metanit.com/sharp/tutorial/.
```

```
\item C\# 5.0 и платформа .NET 4.5:  
    https://professorweb.ru.  
  
\item Уроки программирования с нуля. Си-шарп  
    для чайников:  
    http://mycsharp.ru/.  
  
\item Записная книжка программиста-новичка,  
    C\#, SQL, PHP и все-все-все:  
    http://nullpro.info/.  
\end{enumerate}  
  
\end{document}
```

Оформите ссылки надлежащим образом, преобразовав их в гиперссылки. Дополнительно настройте их свойства (см. рис. 3.41).

Результат 1

Литература

- [1] Валгина, Н.С. Теория текста. Учебное пособие – М.: Логос, 2003. – 173 с.
- [2] Ширинкина, Л.В. Методы восприятия текста / Актуальные проблемы философии, социологии и политологии, экономики и психологии: 22 Материалы юбилейной научной студенческо-аспирантской конференции (апрель 2001)/ Перм.ун-т. – Пермь, 2001. – Вып.4 – с. 138-141.
- [3] Львовский, С.М. Набор и верстка в системе LaTeX / С.М. Львовский. – М.: МЦНМО, 2014. – 398 с.
- [4] Львовский, С.М. Работа в системе LaTeX / Львовский С.М. – М. Национальный Открытый Университет «ИНТУИТ», 2007. – 465 с.
- [5] Котельников, И. А., Чеботаев, П. З. LaTeX2e по-русски. – Новосибирск: Сибирский Хронограф, 2004. – 496 с.

Рис. 3.40. Ожидаемый результат для задания 1.

Результат 2

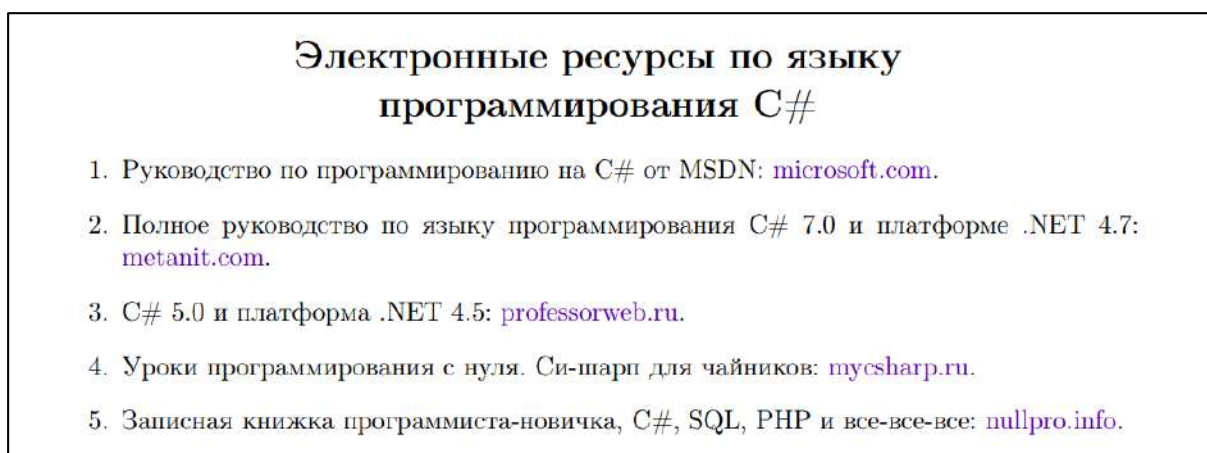


Рис. 3.41. Ожидаемый результат для задания 2.

3.6 Математические тексты

3.6.1 Общие сведения о формулах

Пожалуй, наиболее существенное преимущество LaTeX – высочайшее качество оформления математических символов. В первой главе мы уже отмечали, что LaTeX отлично подходит для верстки технической документации и документов, содержащих математическую символику [45].

Виды формул

Формулы LaTeX можно разделить на две категории:

- внутренние – пишутся в строке абзаца;
- выключенные – указываются отдельной строкой и выравниваются по центру.

Для набора внутренней формулы ее текст помещают между символами

`\(выражение \)`

Например:

Квадратное уравнение имеет вид `\(ax^2+bx+c=0\)`, где `\(a\)`, `\(b\)` и `\(c\)` -- действительные коэффициенты.

Квадратное уравнение имеет вид $ax^2 + bx + c = 0$, где a , b и c – действительные коэффициенты.

Рис. 3.42. Пример оформления внутренних формул (включенных в строку).

Выключенная формула набирается внутри следующих ограничителей:

`\[формула \]`

Например:

Согласно тождеству Эйлера справедливо:

`\[`

$$e^{i\pi} + 1 = 0.$$

`\]`
 где i -- комплексная единица.

Согласно тождеству Эйлера справедливо:

$$e^{i\pi} + 1 = 0,$$

где i – комплексная единица.

Рис. 3.43. Пример оформления выключенной формулы (оформляется отдельным абзацем с выравниванием по центру).

Формулы, в отличие от текста, имеют несколько иной алгоритм обработки. Укажем некоторые особенности:

1. Любые пробелы игнорируются и настраиваются только специальными командами (горизонтальными пробелами).
2. В формулах запрещены пустые строки.
3. Шрифт формулы устанавливается курсивом.

Старый стандарт оформления формул

LaTeX поддерживает и другой, более старый стандарт оформления формул. Так, внутренняя формула помещается между ограничителями

`$формула$`

а выключенная формула –

\$\$
формула
\$\$

На самом деле сейчас нет жесткого требования к выбору нового или старого оформления. Используйте то, которое вам удобнее.

AMS

По умолчанию все типы документов LaTeX поддерживают верстку формул. Однако стандартные возможности достаточно ограничены и не содержат описания многих необходимых элементов оформления.

Поэтому при верстке научных статей, книг и любых других документов крайне полезным будет подключение дополнительных пакетов серии AMS (Американское Математическое Общество). В них содержится описание огромного числа математических и связанных с ними символов, специальных окружений, которые позволяют оформлять определения, теоремы, свойства и другие подобные структуры, в которых нужна автоматическая вложенная нумерация.

Во всех примерах далее предполагается, что указанные ниже пакеты подключены.

```
\usepackage{amsmath, amssymb, amsthm, amsfonts, amscd}
```

Отметим, что пакеты AMS включены во все основные дистрибутивы по LaTeX.

3.6.2 Оформление формул

Верхние и нижние индексы

Команда	Описание
<code>_</code>	Нижний индекс.
<code>^</code>	Верхний индекс.
<code>{ }</code>	Скобки для ограничения выражения/

Пример

Для прямоугольного треугольника с катетами a , b и гипотенузой c справедливо

$$c^2 = a^2 + b^2.$$

Формула серной кислоты --- H_2SO_4 .

Двойной индекс: X_{2017}^{2018} .
Если забыть про скобки: X_{2017}^{2018} .

Результат

Для прямоугольного треугольника с катетами a , b и гипотенузой c справедливо
$c^2 = a^2 + b^2.$
Формула серной кислоты — H_2SO_4 .
Двойной индекс: X_{2017}^{2018} .
Если забыть про скобки: X_{2017}^{2018} .

Рис. 3.44. Пример оформления индексов.

Дроби

`\frac{числитель}{знаменатель}`

Печатает дробь с указанным числителем и знаменателем.

Пример

Рациональное число --- дробь вида

`$$`

`\frac{m}{n},`

`$$`

где $m \in \mathbb{N}$, $n \in \mathbb{Z}$.

С помощью разложения функции в ряд Тейлора можно приближенно

вычислять ее значения в некоторой точке, например:

`$$`

`\cos x \thickapprox`

`1 + \frac{x^2}{2!} - \frac{x^4}{4!}.`

`$$`

Результат

Рациональное число — дробь вида

$$\frac{m}{n},$$

где $m \in \mathbb{Z}$, $n \in \mathbb{N}$.

С помощью разложения функции в ряд Тейлора можно приближенно вычислять ее значения в некоторой точке, например:

$$\cos x \approx 1 + \frac{x^2}{2!} - \frac{x^4}{4!}.$$

Рис. 3.45. Пример оформления дробей.

При вложении дробей в дроби (т.н. «многоэтажные дроби») LaTeX будет их масштабировать. Если требуется этого избежать, ука-

жите команду `\displaystyle`. Примечательно, что она действует только в пределах уровня дроби, на котором указана.

Пример

Дробь автоматически масштабируется:

```
$$  
\frac{1}{1+\frac{1}{x}}  
$$
```

Дробь не масштабируется:

```
$$  
\frac{1}{\displaystyle 1+\frac{1}{x}}  
$$
```

Результат

Дробь автоматически масштабируется:

$$\frac{1}{1 + \frac{1}{x}}$$

Дробь не масштабируется:

$$\frac{1}{1 + \frac{1}{x}}$$

Рис. 3.46. Пример масштабирования дроби.

Корни

```
\sqrt{выражение}  
\sqrt[n]{выражение}
```

*Первая команда берет квадратный корень от выражения.
Вторая команда берет корень n-й степени от выражения.*

Пример

```
$$
```

$f(x) = \sqrt{\frac{x^4+1}{x^2+1}}$
 $\$ \$$

Забавный факт из математики:

$\$ \$$
 $\sqrt{2 + \sqrt{2 + \sqrt{2 + \ldots}}} = 2.$
 $\$ \$$

$\$ \$$
 $\sqrt[3]{8} = 2$
 $\$ \$$

Результат

$$f(x) = \sqrt{\frac{x^4 + 1}{x^2 + 1}}$$

Забавный факт из математики:

$$\sqrt{2 + \sqrt{2 + \sqrt{2 + \ldots}}} = 2.$$

$$\sqrt[3]{8} = 2$$

Рис. 3.47. Оформление корней.

Скобки

Скобки выступают в роли ограничителей операций формулы. По умолчанию LaTeX не масштабирует скобки согласно высоте вложенного в них выражения, поэтому такой вариант оформления подходит только для формул не выше одной строки.

Для масштабирования скобок к открывающей скобке приписывают команду **\left**, а к закрывающей – **\right**:

$\backslash\mathrm{left}(\text{выражение}\ \backslash\mathrm{right})$

Если скобку требуется установить только с одной стороны, то для левого ограничителя используйте команду вида

`\left(` выражение `\right.`

а для правого

`\left.` выражение `\right)`

В следующей таблице приведены поддерживаемые в LaTeX скобки:

Команда	Вид	Команда	Вид
<code>(</code>	$($	<code>)</code>	$)$
<code>[</code>	$[$	<code>]</code>	$]$
<code>\{</code>	$\{$	<code>\}</code>	$\}$
<code>\lfloor</code>	\lfloor	<code>\rfloor</code>	\rfloor
<code>\lceil</code>	\lceil	<code>\rceil</code>	\rceil
<code>\langle</code>	\langle	<code>\rangle</code>	\rangle
<code> </code>	$ $	<code>\ </code>	$\ $
<code>/</code>	$/$	<code>\backslash</code>	\backslash

Рис. 3.48. Виды скобок, доступные в формулах.

Пример

Обычные скобки не масштабируются:

```
$$
(1+\frac{1}{2})
$$
```

А вот так лучше:

```
$$
\left( 1+\frac{1}{2} \right)
$$
```

Непарный ограничитель:

```
$$
|x| =
\left\{
\begin{array}{l}
x, \text{ если } x \geqslant 0, \\
-x, \text{ если } x < 0.
\end{array}
\right.
$$
```

Здесь отметим ряд важных замечаний. Во-первых, для вывода текста в формуле используется команда `\mbox`. Во-вторых, окружение `array` позволяет оформлять матрицы (таблично подобные структуры).

Результат

Обычные скобки не масштабируются:

$$(1 + \frac{1}{2})$$

А вот так лучше:

$$\left(1 + \frac{1}{2}\right)$$

Непарный ограничитель:

$$|x| = \begin{cases} x, & \text{если } x \geq 0, \\ -x, & \text{если } x < 0. \end{cases}$$

Рис. 3.49. Использование парных и непарных ограничителей (скобок).

Символ суммирования

`\sum`

Оператор суммирования.

С помощью верхних и нижних индексов можно проставлять пределы суммирования.

Пример

Среднее арифметическое нескольких чисел:

`$$`

`\overline{X} = \frac{a_1 + a_2 + \ldots + a_n}{n}`

`$$`

или кратко

`$$`

`\overline{X} = \frac{1}{n} \sum_{k=1}^n a_k`

`$$`

Результат

Среднее арифметическое нескольких чисел:

$$\overline{X} = \frac{a_1 + a_2 + \dots + a_n}{n}$$

или кратко

$$\overline{X} = \frac{1}{n} \sum_{k=1}^n a_k$$

Рис. 3.50. Оформление символа суммы (с пределами суммирования).

Символы интегрирования

`\int`, `\iint`, `\iiint`, `\idotsint`, `\oint`

Символы интегралов. Первые три – обычный, двойной, тройной и кратный интеграл, последний – по замкнутому контуру.

Для указания пределов интегрирования используются команды форматирования индексов. Также после команды интеграла может следовать команда **\limits**, которая размечает пределы интегрирования строго сверху и снизу символа интеграла. (В обычном режиме пределы печатаются справа и сбоку, поэтому по вертикали такая формула будет занимать меньше места).

Пример

Неопределенный интеграл:

$$\int (ax + b) \, dx = \frac{ax^2}{2} + bx + c.$$

Определенный интеграл (форма 1):

$$\int_a^b f(x) \, dx.$$

Определенный интеграл (форма 2):

$$\int\limits_a^b f(x) \, dx.$$

Двойной и тройной интеграл:

$$\iint\limits_{\Omega} f(x,y) \, dx \, dy, \quad \iiint\limits_{\Omega} f(x,y,z) \, dx \, dy \, dz.$$

Кратный интеграл в общем виде:

$$\int\limits_{\Omega} f(x) \, dx.$$

Криволинейный интеграл:

$$\oint_C F(x,y) \, dx + G(x,y) \, dy.$$

Результат

Неопределенный интеграл:

$$\int (ax + b) dx = \frac{ax^2}{2} + bx + c.$$

Определенный интеграл (форма 1):

$$\int_a^b f(x) dx.$$

Определенный интеграл (форма 2):

$$\int_a^b f(x) dx.$$

Двойной и тройной интеграл:

$$\iint_{\Omega} f(x, y) dx dy, \quad \iiint_{\Omega} f(x, y, z) dx dy dz.$$

Кратный интеграл в общем виде:

$$\int \cdots \int_{\Omega} f(x) dx.$$

Криволинейный интеграл:

$$\oint_C F(x, y) dx + G(x, y) dy.$$

Рис. 3.51. Оформление интегралов.

Пределы

`\lim`

Печатает символ предела.

Пример

\$\$
\lim_{n \rightarrow \infty} 1 + \frac{1}{n} = 1.
\$\$

Результат

$$\lim_{n \rightarrow \infty} 1 + \frac{1}{n} = 1.$$

Рис. 3.52. Оформление пределов.

Математические функции и греческие буквы

В следующей таблице приведены математические функции. По правилам типографики их принято оформлять прямым начертанием.

Команда	Вид	Команда	Вид	Команда	Вид
<code>\sin</code>	sin	<code>\cos</code>	cos	<code>\tan</code>	tan
<code>\ctg</code>	ctg				
<code>\arcsin</code>	arcsin	<code>\arccos</code>	arccos	<code>\arctan</code>	arctan
<code>\sec</code>	sec	<code>\csc</code>	csc		
<code>\log</code>	log	<code>\ln</code>	ln	<code>\lg</code>	lg
<code>\exp</code>	exp				
<code>\sinh</code>	sinh	<code>\cosh</code>	cosh	<code>\tanh</code>	tanh
<code>\min</code>	min	<code>\max</code>	max	<code>\dim</code>	dim

Рис. 3.53. Команды для оформления математических функций.

Греческие символы также набираются с помощью специальных команд (рис. 3.54):

α	<code>\alpha</code>	λ	<code>\lambda</code>	ϕ	<code>\phi</code>
β	<code>\beta</code>	μ	<code>\mu</code>	χ	<code>\chi</code>
γ	<code>\gamma</code>	ν	<code>\nu</code>	ψ	<code>\psi</code>
δ	<code>\delta</code>	ξ	<code>\xi</code>	ω	<code>\omega</code>
ϵ	<code>\epsilon</code>	\omicron	<code>\omicron</code>	ε	<code>\varepsilon</code>
ζ	<code>\zeta</code>	π	<code>\pi</code>	ϑ	<code>\vartheta</code>
η	<code>\eta</code>	ρ	<code>\rho</code>	ϖ	<code>\varpi</code>
θ	<code>\theta</code>	σ	<code>\sigma</code>	ϱ	<code>\varrho</code>
ι	<code>\iota</code>	τ	<code>\tau</code>	ς	<code>\varsigma</code>
κ	<code>\kappa</code>	υ	<code>\upsilon</code>	φ	<code>\varphi</code>
Γ	<code>\Gamma</code>	Ξ	<code>\Xi</code>	Φ	<code>\Phi</code>
Δ	<code>\Delta</code>	Π	<code>\Pi</code>	Ψ	<code>\Psi</code>
Θ	<code>\Theta</code>	Σ	<code>\Sigma</code>	Ω	<code>\Omega</code>
Λ	<code>\Lambda</code>	Υ	<code>\Upsilon</code>		

Рис. 3.54. Прописные и заглавные греческие буквы.

Пример

Косинус двойного угла:

```


$$\cos 2x = \cos^2 x - \sin^2 x.$$


```

Сложная функция:

```


$$y(x) = \cos \left( \frac{1 + x^2}{\ln|x+1|} \right).$$


```

Греческие символы:

```


$$F(x, \xi, \psi) = \alpha \xi^2(x) + \beta \psi^2(x).$$


```

Результат

Косинус двойного угла:

$$\cos 2x = \cos^2 x - \sin^2 x.$$

Сложная функция:

$$y(x) = \cos \left(\frac{1 + x^2}{\ln |x + 1|} \right).$$

Греческие символы:

$$F(x, \xi, \psi) = \alpha \xi^2(x) + \beta \psi^2(x).$$

Рис. 3.55. Пример оформления функций и греческих букв.

3.6.3 Окружение equation

Автоматическая нумерация

Окружение equation

Оформляет выключенные формулы и автоматически нумерует их в порядке следования.

- *Может быть помечена командой `\label`, что позволяет ссылаться на формулу в тексте.*
- *Ссылаться на формулу можно с помощью команды `\ref`.*
- *Более удобной является ссылка `\eqref` из пакета AMS: она автоматически пишет номер в скобках.*

Пример

Формула Эйлера утверждает, что для любого комплексного (действительного) числа x выполнено следующее равенство:

```
\begin{equation} \label{eq:euler_formula}
e^{ix} = \cos x + i \sin x
\end{equation}
```

В частности, при $x=\pi$ в \eqref{eq:euler_formula}, получим тождество Эйлера:

```
\begin{equation}
e^{i\pi} + 1 = 0.
\end{equation}
```

Результат

Формула Эйлера утверждает, что для любого комплексного (действительного) числа x выполнено следующее равенство:

$$e^{ix} = \cos x + i \sin x \quad (1)$$

В частности, при $x = \pi$ в (1), получим тождество Эйлера:

$$e^{i\pi} + 1 = 0. \quad (2)$$

Рис. 3.56. Пример автоматической нумерации выключенных формул и ссылок на них.

Произвольное обозначение формул

Формула может быть обозначена любым символом или текстом. Для этого в конце формулы используется команда `\eqno`. Эта команда работает только для выключенных формул внутри парных символов `$$`.

Пример

Для комплексной единицы справедливо
`$$`

`i^2 + 1 = 0, \quad \text{где } x \in \mathbb{C}.`
`\eqno(**)`

`$$`

Результат

Для комплексной единицы справедливо

$$i^2 + 1 = 0, \quad \text{где } x \in \mathbb{C}. \quad (**)$$

Рис. 3.57. Замена числового обозначения формулы на произвольное.

Старайтесь избегать такой подход в обозначении формул: он не позволяет организовать автоматически обновляемую ссылку.

Кроме того, старайтесь работать именно с окружением `equation`. Оно обезопасит вас от неверной нумерации формул вручную.

Окружение `equation*`

По существу – это аналог выключенной формулы `$$`. Привязывать метку к такой формуле, очевидно, не имеет смысла. Тем не менее, если вы используете окружение `equation`, то для поддержки единообразия стиля верстки следует использовать и `equation*` вместо `$$`.

3.6.4 Разные символы

Пакеты серии AMS поддерживают большое число разнообразных математических символов. Подробную информацию читатель найдет в справочных материалах. Приведем еще ряд примеров часто используемых символов.

†	<code>\dag</code>	§	<code>\S</code>	£	<code>\pounds</code>
‡	<code>\ddag</code>	¶	<code>\P</code>	№	<code>\No</code>
Ø	<code>\O</code>	ø	<code>\o</code>	©	<code>\copyright</code>
Œ	<code>\OE</code>	œ	<code>\oe</code>	ı	<code>\i</code>
Æ	<code>\AE</code>	æ	<code>\ae</code>	ı	<code>\j</code>
Å	<code>\AA</code>	å	<code>\aa</code>	ß	<code>\ss</code>
Ł	<code>\L</code>	ł	<code>\l</code>		

Рис. 3.58. Специальные символы.

\aleph	<code>\aleph</code>	$'$	<code>\prime</code>	\forall	<code>\forall</code>	<code>\forall</code>	<code>\forall</code>
\hbar	<code>\hbar</code>	\emptyset	<code>\emptyset</code>	\exists	<code>\exists</code>	\exists	<code>\exists</code>
\imath	<code>\imath</code>	∇	<code>\nabla</code>	\neg	<code>\neg</code>	\neg	<code>\neg</code>
\jmath	<code>\jmath</code>	\surd	<code>\surd</code>	\flat	<code>\flat</code>	\flat	<code>\flat</code>
ℓ	<code>\ell</code>	\top	<code>\top</code>	\natural	<code>\natural</code>	\natural	<code>\natural</code>
\wp	<code>\wp</code>	\perp	<code>\bot</code>	\sharp	<code>\sharp</code>	\sharp	<code>\sharp</code>
\Re	<code>\Re</code>	\parallel	<code>\Vert</code>	\clubsuit	<code>\clubsuit</code>	\clubsuit	<code>\clubsuit</code>
\Im	<code>\Im</code>	\angle	<code>\angle</code>	\diamondsuit	<code>\diamondsuit</code>	\diamondsuit	<code>\diamondsuit</code>
∂	<code>\partial</code>	\triangle	<code>\triangle</code>	\heartsuit	<code>\heartsuit</code>	\heartsuit	<code>\heartsuit</code>
∞	<code>\infty</code>	\backslash	<code>\backslash</code>	\spadesuit	<code>\spadesuit</code>	\spadesuit	<code>\spadesuit</code>
\mho	<code>\mho</code>	\square	<code>\Box</code>	\Diamond	<code>\Diamond</code>	\Diamond	<code>\Diamond</code>
\cdots	<code>\cdots</code>	\vdots	<code>\vdots</code>	\ddots	<code>\ddots</code>	\ddots	<code>\ddots</code>
\angle	<code>\angle</code>	\beth	<code>\beth</code>	\square	<code>\square</code>	\square	<code>\square</code>
\measuredangle	<code>\measuredangle</code>	\gimel	<code>\gimel</code>	\blacksquare	<code>\blacksquare</code>	\blacksquare	<code>\blacksquare</code>
\sphericalangle	<code>\sphericalangle</code>	\daleth	<code>\daleth</code>	\lozenge	<code>\lozenge</code>	\lozenge	<code>\lozenge</code>
\Finv	<code>\Finv</code>	\digamma	<code>\digamma</code>	\blacklozenge	<code>\blacklozenge</code>	\blacklozenge	<code>\blacklozenge</code>
\Game	<code>\Game</code>	\eth	<code>\eth</code>	\triangledown	<code>\triangledown</code>	\triangledown	<code>\triangledown</code>
\mho	<code>\mho</code>	\Bbbk	<code>\Bbbk</code>	\blacktriangledown	<code>\blacktriangledown</code>	\blacktriangledown	<code>\blacktriangledown</code>
\nexists	<code>\nexists</code>	\hslash	<code>\hslash</code>	\blacktriangle	<code>\blacktriangle</code>	\blacktriangle	<code>\blacktriangle</code>
\varnothing	<code>\varnothing</code>	\hbar	<code>\hbar</code>	\bigstar	<code>\bigstar</code>	\bigstar	<code>\bigstar</code>
\textcircled{S}	<code>\textcircled{S}</code>	\backprime	<code>\backprime</code>	\complement	<code>\complement</code>	\complement	<code>\complement</code>

Рис. 3.59. Разнообразные символы.

$+$	<code>+</code>	\oplus	<code>\oplus</code>	\setminus	<code>\setminus</code>	\setminus	<code>\setminus</code>
$-$	<code>-</code>	\ominus	<code>\ominus</code>	$/$	<code>/</code>	$/$	<code>/</code>
\pm	<code>\pm</code>	\otimes	<code>\otimes</code>	\vee	<code>\vee</code>	\vee	<code>\vee</code>
\mp	<code>\mp</code>	\oslash	<code>\oslash</code>	\wedge	<code>\wedge</code>	\wedge	<code>\wedge</code>
\times	<code>\times</code>	\odot	<code>\odot</code>	\cap	<code>\cap</code>	\cap	<code>\cap</code>
\div	<code>\div</code>	\bigcirc	<code>\bigcirc</code>	\cup	<code>\cup</code>	\cup	<code>\cup</code>
$*$	<code>\ast</code>	\dagger	<code>\dagger</code>	\uplus	<code>\uplus</code>	\uplus	<code>\uplus</code>
\star	<code>\star</code>	\ddagger	<code>\ddagger</code>	\sqcap	<code>\sqcap</code>	\sqcap	<code>\sqcap</code>
\diamond	<code>\diamond</code>	\wr	<code>\wr</code>	\sqcup	<code>\sqcup</code>	\sqcup	<code>\sqcup</code>
\circ	<code>\circ</code>	\bullet	<code>\bullet</code>	\amalg	<code>\amalg</code>	\amalg	<code>\amalg</code>
\triangleleft	<code>\triangleleft</code>	\cdot	<code>\cdot</code>	\bigtriangleup	<code>\bigtriangleup</code>	\bigtriangleup	<code>\bigtriangleup</code>
\triangleright	<code>\triangleright</code>	$:$	<code>:</code>	\bigtriangledown	<code>\bigtriangledown</code>	\bigtriangledown	<code>\bigtriangledown</code>

Рис. 3.60. Символы операторов.

$<$	<code><</code>	$>$	<code>></code>	$=$	<code>=</code>
\leq	<code>\le</code>	\geq	<code>\ge</code>	\equiv	<code>\equiv</code>
\prec	<code>\prec</code>	\succ	<code>\succ</code>	\sim	<code>\sim</code>
\preceq	<code>\preceq</code>	\succeq	<code>\succeq</code>	\simeq	<code>\simeq</code>
\ll	<code>\ll</code>	\gg	<code>\gg</code>	\approx	<code>\approx</code>
\subset	<code>\subset</code>	\supset	<code>\supset</code>	\cong	<code>\cong</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\doteq	<code>\doteq</code>
\sqsubset	<code>\sqsubset</code>	\sqsupset	<code>\sqsupset</code>	\asymp	<code>\asymp</code>
\sqsubseteq	<code>\sqsubseteq</code>	\sqsupseteq	<code>\sqsupseteq</code>	\bowtie	<code>\bowtie</code>
\in	<code>\in</code>	\ni	<code>\ni</code>	\Join	<code>\Join</code>
\smile	<code>\smile</code>	\frown	<code>\frown</code>	\vdash	<code>\vdash</code>
\mid	<code>\mid</code>	\parallel	<code>\parallel</code>	\dashv	<code>\dashv</code>
\propto	<code>\propto</code>	\perp	<code>\perp</code>	\models	<code>\models</code>
$\not<$	<code>\not<</code>	$\not>$	<code>\not></code>	\neq	<code>\neq</code>
$\not\leq$	<code>\not\le</code>	$\not\geq$	<code>\not\ge</code>	$\not\equiv$	<code>\not\equiv</code>
$\not\prec$	<code>\not\prec</code>	$\not\succ$	<code>\not\succ</code>	$\not\sim$	<code>\not\sim</code>
$\not\preceq$	<code>\not\preceq</code>	$\not\succeq$	<code>\not\succeq</code>	$\not\simeq$	<code>\not\simeq</code>
$\not\subset$	<code>\not\subset</code>	$\not\supset$	<code>\not\supset</code>	$\not\approx$	<code>\not\approx</code>
$\not\subseteq$	<code>\not\subseteq</code>	$\not\supseteq$	<code>\not\supseteq</code>	$\not\cong$	<code>\not\cong</code>
$\not\sqsubseteq$	<code>\not\sqsubseteq</code>	$\not\sqsupseteq$	<code>\not\sqsupseteq</code>	$\not\asymp$	<code>\not\asymp</code>

Рис. 3.61. Операторы отношений.

\uparrow	<code>\uparrow</code>	\leftarrow	<code>\leftarrow</code>	\longleftarrow	<code>\longleftarrow</code>
\Uparrow	<code>\Uparrow</code>	\Leftarrow	<code>\Leftarrow</code>	\Longleftarrow	<code>\Longleftarrow</code>
\downarrow	<code>\downarrow</code>	\rightarrow	<code>\rightarrow</code>	\longrightarrow	<code>\longrightarrow</code>
\Downarrow	<code>\Downarrow</code>	\Rightarrow	<code>\Rightarrow</code>	\Longrightarrow	<code>\Longrightarrow</code>
\updownarrow	<code>\updownarrow</code>	\leftrightarrow	<code>\leftrightarrow</code>	\longleftrightarrow	<code>\longleftrightarrow</code>
\Updownarrow	<code>\Updownarrow</code>	\Leftrightarrow	<code>\Leftrightarrow</code>	\Longleftrightarrow	<code>\Longleftrightarrow</code>
\nearrow	<code>\nearrow</code>	\mapsto	<code>\mapsto</code>	\longmapsto	<code>\longmapsto</code>
\searrow	<code>\searrow</code>	\hookleftarrow	<code>\hookleftarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>
\swarrow	<code>\swarrow</code>	\leftharpoonup	<code>\leftharpoonup</code>	\rightharpoonup	<code>\rightharpoonup</code>
\nwarrow	<code>\nwarrow</code>	\leftharpoondown	<code>\leftharpoondown</code>	\rightharpoondown	<code>\rightharpoondown</code>
\rightsquigarrow	<code>\rightsquigarrow</code>	\rightleftharpoons	<code>\rightleftharpoons</code>	\rightleftharpoons	<code>\rightleftharpoons</code>

Рис. 3.62. Стрелки.

Вопросы для самопроверки

1. Чем отличается внутренняя формула от выключенной?
2. Укажите правила оформления внутренних и выключенных формул.
3. Зачем нужны пакеты серии AMS?
4. Перечислите команды для оформления индексов, дробей, корней.
5. Как правильно оформить скобки?
6. Какие интегралы можно корректно отобразить в LaTeX?
7. Укажите преимущества использования окружения equation.

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Набор математических формул

Оформите документ, изображенный на рис. 3.63-рис. 3.64. Воспользуйтесь следующим шаблоном:

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}
\usepackage[top=2.5cm, bottom=2cm, left=3cm,
right=1.5cm]{geometry}
\usepackage{indentfirst}

\linespread{1.35}
\parindent = 1cm

\begin{document}

    % Код разметки документа

\end{document}
```

Результат 1

1 Работа с индексами

Формула серной кислоты: H_2SO_4 .

Квадратичная функция $y(x) = ax^2 + bx + c$.

Полином степени $n \in \mathbb{N}$:

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0.$$

Очень большое число:

$$10^{200^{3000}}$$

2 Дроби

Аликувотная дробь:

$$\frac{2}{3} = \frac{1}{2} + \frac{1}{3} = \frac{1}{4} + \frac{1}{4} + \frac{1}{6}.$$

Масштабируемое оформление дроби:

$$\frac{1 + \frac{x}{2}}{\frac{y^2}{3}}$$

Немасштабируемое оформление дроби:

$$\frac{1 + \frac{x}{2}}{\frac{y^2}{3}}$$

3 Корни, функции, греческие символы

Для $x \geq 1$ и $n \in \mathbb{N}$, $n \geq 2$, справедливо:

$$\sqrt{x} > \sqrt[n]{x} > \sqrt[n-1]{x} > \dots > \sqrt[2]{x}.$$

Сложная функция:

$$F(\omega, x) = \sqrt{\frac{|\sin(\omega x)| + 1}{3e^{-x}}}.$$

4 Скобки

Вычислить:

$$\left(1 + \left[\frac{3x}{5} + \frac{x}{2}\right]\right)^2,$$

где x – целое неотрицательное число (квадратные скобки означают взятие целой части).

Рис. 3.63. Ожидаемый результат задания 1 (стр. 1).

5 Суммирование и интегрирование

Гёльдеровы нормы n -мерных векторов:

$$\|x\|_p = \left(\sum_k |x_k|^p \right)^{\frac{1}{p}}.$$

Теорема Стокса. Пусть на ориентируемом многообразии M размерности n заданы ориентируемое p -мерное подмногообразие σ и дифференциальная форма ω степени $p - 1$ класса C^1 ($1 \leq p \leq n$). Тогда, если граница подмногообразия $\partial\sigma$ положительно ориентирована, то

$$\int_{\sigma} d\omega = \int_{\partial\sigma} \omega,$$

где $d\omega$ обозначает внешний дифференциал формы ω .

Рис. 3.64. Ожидаемый результат задания 1 (стр. 2).

2. Окружение equation

Изучите структуру следующего документа:

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}
\usepackage[top=2.5cm, bottom=3cm, left=3cm,
right=1.5cm]{geometry}
\usepackage{indentfirst}

% Пакет для настройки отступов у списков
\usepackage{enumitem}
\setlist{nolistsep, labelsep=0.4cm, leftmargin=1.85cm}

\linespread{1.15}
\parindent = 0.75cm

\begin{document}

  \begin{center}
    \huge \textbf{Квадратное уравнение}
  \end{center}

  \section{Определение}
  \textbf{Квадратное уравнение} --- это уравнение
  вида
  \begin{equation}
    ax^2 + bx + c = 0,
  \end{equation}
  где коэффициенты  $a$ ,  $b$  и  $c$  --- произвольные
  числа, причем  $a \neq 0$ .

  Все квадратные уравнения условно можно разбить на
  три класса:
  \begin{enumerate}
    \item имеют один корень;
    \item имеют два различных корня;
    \item не имеют корней.
  \end{enumerate}

  \section{Дискриминант}
```

Как определить, сколько корней имеет квадратное уравнение (1)?

Чтобы это определить, нужно вычислить специальную величину ---\textbf{дискриминант}. Дискриминантом называют число

```
\begin{equation}
D = b^2 - 4ac.
\end{equation}
```

Мы не будем рассматривать, каким образом получена эта величина. Главное, что по знаку дискриминанта можно определить, сколько корней имеет квадратное уравнение. А именно:

```
\begin{enumerate}
\item если  $D < 0$ , корней нет;
\item если  $D = 0$ , есть ровно один корень;
\item если  $D > 0$ , то уравнение (1) имеет
      два различных корня.
\end{enumerate}
```

Например, следующее квадратное уравнение согласно формуле (2) имеет одно решение:

```
\begin{equation*}
x^2 + 10x + 25 = 0.
\end{equation*}
```

```
\section{Вычисление корней}
```

Итак, корни уравнения (1) искать имеет смысл только в случае $D \geq 2$.

Для вычисления корней используются следующие формулы:

```
\begin{equation}
x_1 = \frac{-b - \sqrt{D}}{2a},
\quad
x_2 = \frac{-b + \sqrt{D}}{2a}.
\end{equation}
```

Нетрудно заметить, что в случае $D = 0$ оба корня будут одинаковы, что подтверждает ранее сказанное: при $D = 0$ уравнение (1) имеет ровно один корень. В этом случае используется упрощенная формула:

```
\begin{equation}
x = -\frac{b}{2a}.
\end{equation}
```

```
\section{Неполные квадратные уравнения}
```

Решение квадратного уравнения (1) может быть существенно упрощено, если один из коэффициентов b или c (либо оба вместе) равны нулю. В этом случае вычислять дискриминант и использовать формулы (3), (4) вовсе необязательно!

Например, для уравнения $4x^2 + x = 0$ ($c=0$) осуществим вынос x за скобки:

```
\begin{equation*}
  4x^2 + x = 0
  \quad\quad\quad\rightarrow\quad
  x(4x + 1) = 0
  \quad\quad\quad\rightarrow\quad
  x = 0 \quad\quad\quad\text{или}\quad\quad\quad x = -\frac{1}{4}.
\end{equation*}
```

Другой пример: $5x^2 + 2 = 0$ (где $b = 0$).

```
\begin{equation*}
  5x^2 + 2 = 0
  \quad\quad\quad\rightarrow\quad
  x^2 = -\frac{2}{5}.
\end{equation*}
```

Очевидно, что решения не имеет корней, т.к. левая часть всегда неотрицательная, а правая --- отрицательна.

```
\end{document}
```

С помощью команд перекрестных ссылок оформите ссылки на формулы надлежащим образом (сейчас они обозначены обычным текстом). Результат, на который следует ориентироваться, изображен на рис. 3.65 и рис. 3.66.

Результат 2

Квадратное уравнение

1 Определение

Квадратное уравнение — это уравнение вида

$$ax^2 + bx + c = 0, \quad (1)$$

где коэффициенты a , b и c — произвольные числа, причем $a \neq 0$.

Все квадратные уравнения условно можно разбить на три класса:

1. имеют один корень;
2. имеют два различных корня;
3. не имеют корней.

2 Дискриминант

Как определить, сколько корней имеет квадратное уравнение (1)? Для этого нужно вычислить специальную величину — **дискриминант**. Дискриминантом называют число

$$D = b^2 - 4ac. \quad (2)$$

Мы не будем рассматривать, каким образом получена эта величина. Главное, что по знаку дискриминанта можно определить, сколько корней имеет квадратное уравнение. А именно:

1. если $D < 0$, корней нет;
2. если $D = 0$, есть ровно один корень;
3. если $D > 0$, то уравнение (1) имеет два различных корня.

Например, следующее квадратное уравнение согласно формуле (2) имеет одно решение:

$$x^2 + 10x + 25 = 0.$$

3 Вычисление корней

Итак, корни уравнения (1) искать имеет смысл только в случае $D \geq 0$.

Для вычисления корней используются следующие формулы:

$$x_1 = \frac{-b - \sqrt{D}}{2a}, \quad x_2 = \frac{-b + \sqrt{D}}{2a}. \quad (3)$$

Рис. 3.65. Ожидаемый результат для задания 2 (стр. 1).

Нетрудно заметить, что в случае $D = 0$ оба корня будут одинаковы, что подтверждает ранее сказанное: при $D = 0$ уравнение (1) имеет ровно один корень. В этом случае используется упрощенная формула:

$$x = -\frac{b}{2a}. \quad (4)$$

4 Неполные квадратные уравнения

Решение квадратного уравнения (1) может быть существенно упрощено, если один из коэффициентов b или c (либо оба вместе) равны нулю. В этом случае вычислять дискриминант и использовать формулы (3), (4) вовсе необязательно!

Например, для уравнения $4x^2 + x = 0$ ($c = 0$) осуществим вынос x за скобки:

$$4x^2 + x = 0 \Leftrightarrow x(4x + 1) = 0 \Leftrightarrow x = 0 \text{ или } x = -\frac{1}{4}.$$

Другой пример: $5x^2 + 2 = 0$ (где $b = 0$).

$$5x^2 + 2 = 0 \Leftrightarrow x^2 = -\frac{2}{5}.$$

Очевидно, что у уравнения нет корней, т.к. левая часть всегда неотрицательная, а правая — отрицательна.

Рис. 3.66. Ожидаемый результат для задания 2 (стр. 2).

3.7 Создание новых команд

3.7.1 Макрос `\newcommand`

Необходимость создания новых команд

При наборе большого объема текста вы наверняка столкнетесь с необходимостью неоднократного повтора определенной последовательности команд. Естественное желание в этом случае – каким-либо автоматизировать набор этой части разметки, либо, по крайней мере, уменьшить время ее набора.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}

\begin{document}

  \vspace{0.5em}\noindent\hrulefill\par\vspace{1em}
  \textbf{Задача 1.} Напишите программу, вычисляющую
  наименьшее общее кратное двух заданных целых чисел.

  \vspace{0.5em}\noindent\hrulefill\par\vspace{1em}
  \textbf{Задача 2.} Напишите программу, выводящую на
  экран пары дружественных чисел. Напомним, что два
  числа называют дружественными, если сумма делителей
  первого числа равна второму числу, а сумма делителей
  второго числа равна первому числу (само число в сумму
  делителей не включается)

  \vspace{0.5em}\noindent\hrulefill\par\vspace{1em}
  \textbf{Задача 3.} Используя ...

\end{document}
```

Результат

Задача 1. Напишите программу, вычисляющую наименьшее общее кратное двух заданных целых чисел.

Задача 2. Напишите программу, выводящую на экран пары дружественных чисел. Напомним, что два числа называют дружественными, если сумма делителей первого числа равна второму числу, а сумма делителей второго числа равна первому числу (само число в сумму делителей не включается)

Задача 3. Используя ...

Рис. 3.67. Пример разметки, в которой многократно используется группа одинаковых команд.

В приведенном примере для изображения линии-разделителя с дополнительными отступами приходится каждый раз повторно набирать или копировать следующую последовательность команд:

```
\vspace{0.5em}\noindent\hrulefill\par\vspace{1em}
```

К счастью, LaTeX позволяет изящно решить указанную проблему. Для этого потребуется специальный макрос `\newcommand`.

Описание макроса `\newcommand`

```
\newcommand{команда}{описание команды}
```

Макрос создает новую команду, осуществляющую все указанные операции в описании. Первый параметр не должен совпадать с именем существующих команд LaTeX.

Иными словами, в первом параметре пользователь пишет желаемое название команды (обязательно начинается со знака «\»), а во втором параметре — текст, команду или последовательность текста и команд, которые будут подставляться при написании созданной команды.

Обозначим линию-разделитель из примера командой `\longline`. В преамбуле документа напишем следующее:

```
\newcommand{\longline}{\vspace{0.5em}\noindent\hrulefill\par\vspace{1em}}
```

Теперь все описания линий в тексте можно заменить созданной командой:

```
\longline
\textbf{Задача 1.} Напишите программу, вычисляющую
наименьшее общее кратное двух целых чисел.
```

```
\longline
\textbf{Задача 2.} Напишите программу, выводящую на экран
пары дружественных чисел. Напомним, что два числа
называют дружественными, если сумма делителей первого
числа равна второму числу, а сумма делителей второго
числа равна первому числу (само число в сумму делителей
не включается)
```

```
\longline
\textbf{Задача 3.} Используя ...
```

Оптимизация получилась впечатляющей! А код разметки стал даже проще для чтения и понимания.

Почему макрос?

У читателя может возникнуть вопрос: почему команда `\newcommand` называется макросом, а не просто командой? Для ответа на этот вопрос проще всего узнать, что понимают под словом «макрос».

В программировании **макросом** называют символьное имя, закрепленное за некоторой последовательностью программных инструкций. Это имя не является отдельной инструкцией само по себе, оно лишь нужно программисту для удобного обращения к серии подряд идущих инструкций, которые часто повторяются в коде. Во время работы компилятора вместо имени макроса подставляются связанные с ним инструкции.

Точно такую же функцию выполняет и макрос `\newcommand`. Это не удивительно: ранее мы указывали, что TeX и LaTeX можно рассматривать не только в качестве языков разметки, но и языков с элементами программирования.

Работа с параметрами

На самом деле, макрос `\newcommand` весьма гибок в своих возможностях. Поскольку `LaTeX` имеет команды с параметрами, то `\newcommand` также можно настроить на их использование.

```
\newcommand{команда}[число_параметров]{описание команды}
```

Макрос создает новую команду, осуществляющую все указанные в описании операции. В опциях указывается число параметров макроса. В описании макроса для подстановки значения соответствующего параметра пишут `#1`, `#2`, и т.д.

Расширим возможности нашей команды `\longline`: пусть интервал до и после линии можно указать произвольно, через параметры команды. Исправим макрос следующим образом:

```
\newcommand{\longline}[2]{\vspace{#1}\noindent\hrulefill\par\vspace{#2}}
```

В квадратных скобках указано число параметров, а вместо `#1` и `#2` будут подставляться указанные нами значения. Например

```
\longline{1em}{2.5em}
```

будет эквивалентна командам

```
\vspace{1em}\noindent\hrulefill\par\vspace{2.5em}
```

Новые команды можно использовать и для упрощения набора формул. Например, удобным может оказаться макрос для набора определенного интеграла:

```
\newcommand{\Integral}[3]{\int\limits_{#1}^{#2} #3 \: dx}
```

Следующие формулы дают эквивалентный результат:

```
$$
\int\limits_{a}^{b} \sin x \: dx \quad \% \text{обычный набор}
$$
$$
\Integral{a}{b}{\sin x} \quad \% \text{с макросом}
$$
```

Как ни парадоксально, однако есть ситуация, когда использование макросов может быть нежелательно. Так, если вы пишете статью для сборника, то редакционный отдел может не согласиться принять вашу работу с макросами, либо потребовать дополнительную плату за корректировку. Это связано с тем, что в больших сборниках часто попадают макросы с одинаковыми именами, но от разных авторов. Разумеется, в рамках одного сборника они будут конфликтовать между собой.

Поэтому старайтесь использовать макросы только в тех работах, которые предназначены для вас и ваших коллег.

3.7.2 Новые окружения

Помимо команд, пользователь LaTeX может создавать и собственные окружения.

Команда `\newenvironment`

```
\newenvironment{название}[число_параметров]
{оформление_до}{оформление_после}
```

Команда создает новое окружение с указанным названием. Оформление до и после предполагает связанные команды, указывающие начало и конец оформления. Команда может работать с дополнительными параметрами, число которых указывается в опциях.

В отличие от макроса `\newcommand`, новое окружение берет на себя одновременно и функцию команды, и функцию бокса, оформляющего указанный в нем фрагмент.

Проще всего понять работу окружения на примере.

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
```

```

\usepackage[english,russian]{babel}

\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}

% Новое окружение
\newenvironment{nicetext}
{ \bfseries\large \rule{1ex}{1ex}\hfill }
{ \hfill\rule{1ex}{1ex} \vspace{1em} }

\begin{document}

    \begin{nicetext}
        Симпатичный текст
    \end{nicetext}

\end{document}

```

Результат

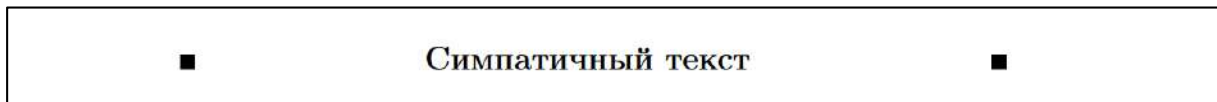


Рис. 3.68. Создание нового окружения для оформления фрагмента текста.

В приведенном примере открывающий блок устанавливает полужирное начертание и увеличенный размер текста, команда `\rule` рисует закрашенный прямоугольник заданной высоты и ширины. Закрывающий блок также рисует прямоугольник и делает отступ снизу. Команды `\hfill` расталкивают содержимое справа и слева, что равносильно выравниванию по центру. Содержимое окружения вставляется между открывающим и закрывающим блоком в процессе компиляции документа. Заметим, что созданное окружение качественно работает только для небольшого текста не более чем в одну строку.

Команда `\newenvironment` с параметрами

Расширить действие окружения можно применив его с параметрами.

В следующем примере создается окружение, оформляющее блоки с описанием примеров (внимательно разберите роль каждой команды):

Пример

```
\documentclass[12pt]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}

% Новое окружение
\newenvironment{example}[1]
{
    \noindent \hrulefill
    \vspace{1em}
    { \centering \large \textbf{Пример #1} \vspace{1em}
\par \normalsize }
}
{ \par \noindent \hrulefill } % закрывающий блок

\begin{document}

    \begin{example}{10}
        Текст примера
    \end{example}

    \begin{example}{<<Числа Фибоначчи>>}
        Текст примера
    \end{example}

\end{document}
```

Результат

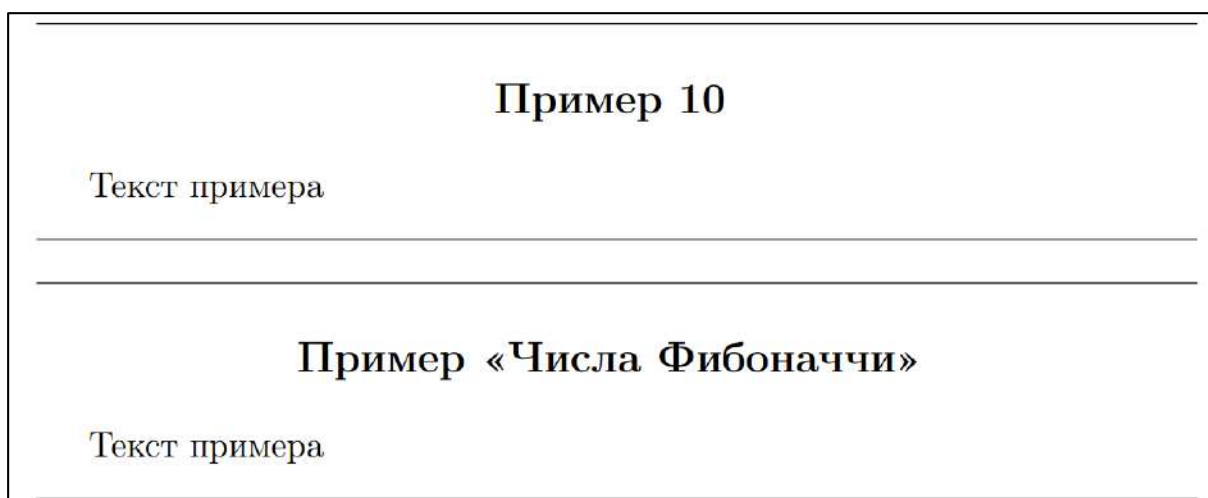


Рис. 3.69. Создание нового окружения с одним параметром.

Очевидно, что алгоритм работы с параметрами такой же, как и у макроса `\newcommand`.

Вопросы для самопроверки

1. Приведите примеры ситуаций, когда использование макроса `\newcommand` может упростить работу.
2. Почему `\newcommand` называют макросом?
3. В каком случае создание новых команд нецелесообразно?
4. В чем отличие `\newcommand` от `\newenvironment`?
5. Приведите примеры использования окружений.

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Создание новых команд

1. Создайте команду `\bigpar`, заменяющую следующую последовательность команд:

`\vspace{0.5cm}\par`

2. Создайте команду `\bolditalic` с одним параметром, которая делает текст жирным и курсивом одновременно. Текст указывается в параметре, например:

`\bolditalic{Жирный и курсив}`

3. Создайте команду `\GaussFun` с двумя параметрами μ и σ , оформляющую функцию вида:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\cdot\sigma^2}}$$

4. Например, следующая разметка

`$$f(x) = \backslashGaussFun{\mu}{\sigma}$$`
`$$f(x) = \backslashGaussFun{2}{5}$$`

дает результат

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\cdot\sigma^2}}$$

$$f(x) = \frac{1}{5\sqrt{2\pi}} e^{-\frac{(x-2)^2}{2\cdot5^2}}$$

2. Создание новых окружений

Создайте окружение `\bigtitle`. Его задача на отдельном листе распечатать текст жирным начертанием и заглавными буквами с выравниванием по центру (рис. 3.70, рис. 3.71). У команды будет один параметр, указывающий отступ сверху (вам потребуется команда `\vspace*{}`, поскольку команда `\vspace{}` не сделает отступ сверху, если это первая строка).

Вызов команды осуществляется следующим образом:

```
\begin{bigtitle}{5cm}
  Основы работы в \LaTeX
\end{bigtitle}
```

```
\begin{bigtitle}{4cm}
  Веб-программирование с использованием языка\
  JavaScript
\end{bigtitle}
```

Результат

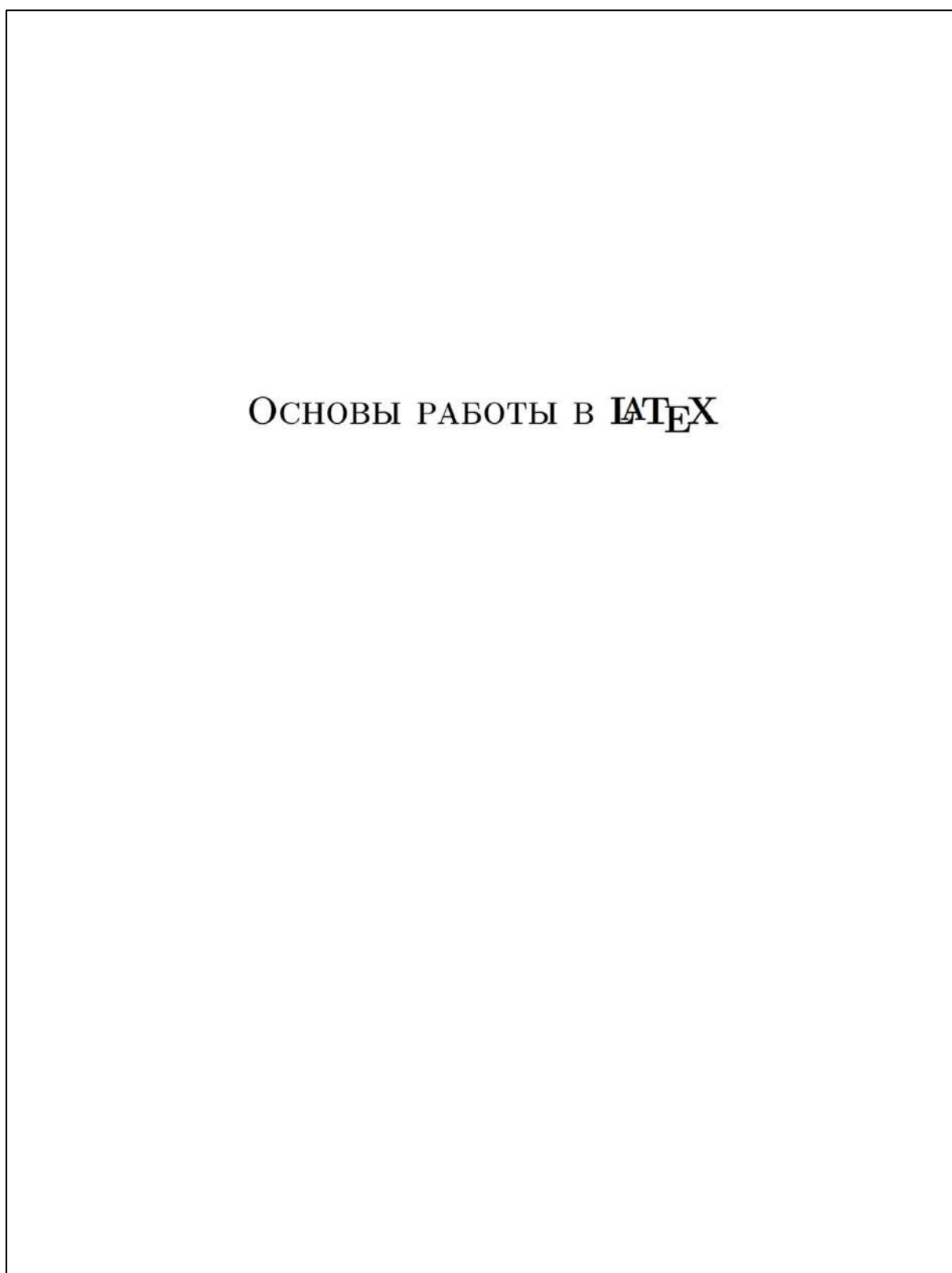


Рис. 3.70. Ожидаемый результат для задания 2 (стр. 1).

Результат

ВЕБ-ПРОГРАММИРОВАНИЕ С
ИСПОЛЬЗОВАНИЕМ ЯЗЫКА
JAVASCRIPT

Рис. 3.71. Ожидаемый результат для задания 2 (стр. 2).

3.8 Презентации

3.8.1 Стандартные средства разработки презентаций

Роль электронной презентации в обучении

Качественно разработанная презентация способна улучшить восприятие и понимание информации, ускорить учебный процесс и повысить мотивацию слушателя. Презентация сегодня – неотъемлемая часть любой научной конференции [16].

К настоящему времени психологами проведено большое число исследований, направленных на изучение особенностей обработки человеком визуальной информации, сформулированы рекомендации по повышению эффективности этой работы.

Учебная презентация должна удовлетворять следующим базовым требованиям:

- Материал презентации должен быть четко структурирован; отдельное внимание должно уделяться навигации в документе.
- Цветовая гамма должна подбираться согласно общим особенностям восприятия человека.
- Информация должна излагаться лаконично и доступно.
- Анимация используется при необходимости и не должна создавать эффект «шума».
- Презентация не должна повторять речь докладчика.
- Элементы презентации должны быть четко различимы.
- Текст должен быть заменен изображением, схемой, диаграммой, таблицей там, где это возможно.
- Выравнивание основного текста рекомендуется делать по левому краю (это связано с тем, что многие офисные пакеты не поддерживают переносы слов в презентациях, или возможны большие пробелы между словами).

Разумеется, этими правилами разработка презентаций не ограничивается. Подробнее рекомендации по разработке презентаций можно найти в [16].

Базовые классы для работы с презентациями

Класс **slides** является одним из первых в LaTeX, который предназначен для создания слайдов. В основе пакета лежит простая идея: информация разбивается на последовательность слайдов. И чтобы у докладчика не было соблазна уместить на слайд как можно больше текста, размер шрифта увеличен [34].

Каждый слайд создается с помощью окружения **slide**.

Пример

```
\documentclass{slides}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}

\usepackage[landscape,left=3cm,right=3cm]{geometry}
\usepackage{ragged2e} % пакет для выравнивания текста
```

```
\begin{document}
  \raggedright % обрывать текст справа

  \begin{slide}
    \begin{center}
      \Huge \textbf{Формула Эйлера}
    \end{center}
  \end{slide}
```

```
\begin{slide}
  \begin{center}
    \large \textbf{Формула Эйлера}
  \end{center}
```

Формула Эйлера связывает комплексную экспоненту с тригонометрическими функциями. Названа в честь Леонарда Эйлера.

Формула Эйлера утверждает, что для любого $x \in \mathbb{C}$

$$e^{i\pi x} = \cos x + i \sin x$$

```
\end{slide}
```

```
\end{document}
```

Результат

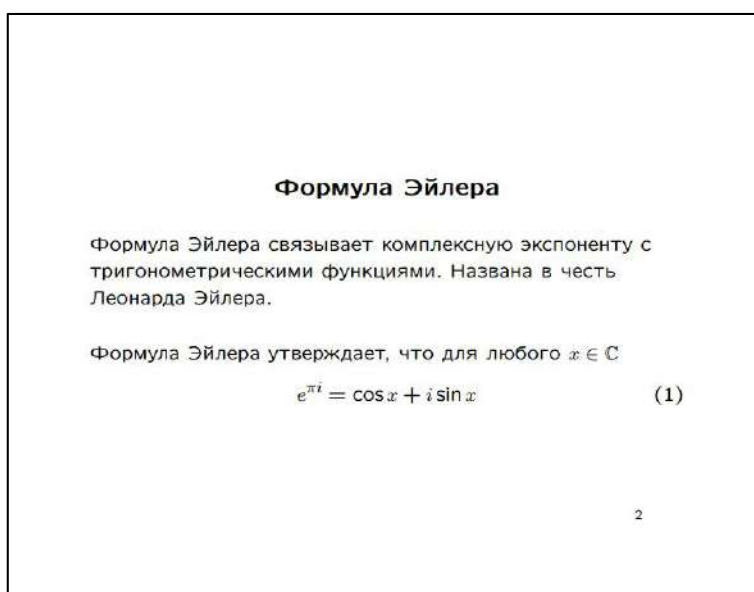
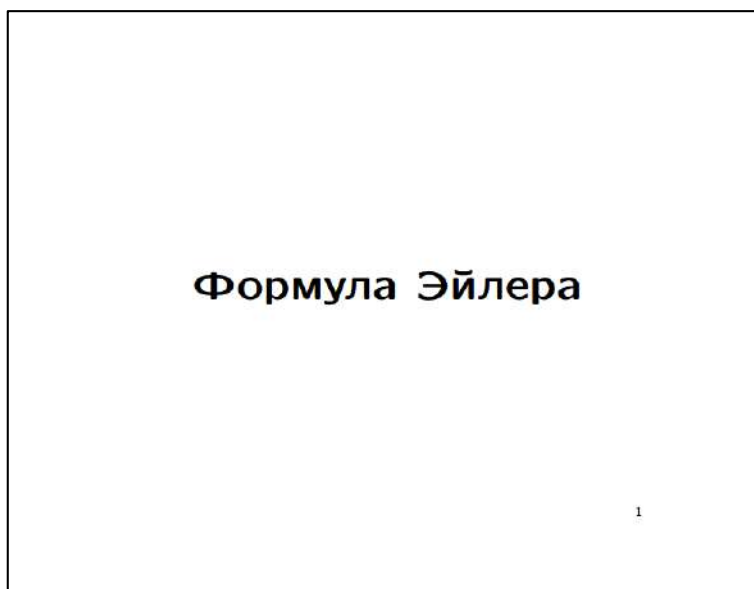


Рис. 3.72. Оформление презентации с помощью класса slides.

В 1993 году появился чуть более современный пакет **seminar**. Однако ни slides, ни seminar не могут обеспечить инструментов для верстки действительно визуально красивых презентаций. Пожалуй, они лишь подойдут в том случае, когда нужно быстро сгенерировать набор слайдов для какого-либо совещания или брифинга.

3.8.2 Пакет Beamer

Существенное развитие механизмов для работы с презентациями произошло с появлением в 2003 году пакета **Beamer**. У пользователей LaTeX появились мощные средства для разработки качественных презентаций, выдержанных в академическом стиле, с возможностью выбора готовых шаблонов или их самостоятельной настройки, элементов анимации и навигации [34,35].

Введение в Beamer

Beamer-презентация делится на набор **фреймов**. Каждый фрейм включает не менее одного слайда (**слоя**), содержащих связанную информацию, частично меняющуюся от слайда к слайду. Такая структура позволяет организовать эффекты анимации в итоговом PDF-документе. В простейшем случае фрейм является слайдом.

Каждый фрейм в общем виде описывается окружением frame.

```
\begin{frame}<слои>[слои_по_умолчанию][параметры]
{заголовок}{подзаголовок}
    Содержание фрейма
\end{frame}
```

Окружение создает фрейм.

Нумерация слоев фрейма начинается с 1. Параметр <слои> указывает слои, которые следует отобразить во фрейме. Если этот параметр не указан, то слои выбираются автоматически на основе параметра <слои по умолчанию> (если первый необязательный аргумент окружения frame заключён в треугольные скобки, то это <слои по умолчанию>, иначе это аргумент параметра).

Параметры для оформления заголовка и подзаголовка можно заменить командами:

```
\frametitle{заголовок}
\framesubtitle{подзаголовок}
```

Значения **t**, **c**, **b** в параметрах задают вертикальное выравнивание содержимого слайда (по верхнему краю, центру и нижнему краю соответственно). По умолчанию используется параметр c.

С помощью команды **\label** для слайда можно создать метку и ссылаться на нее из других слайдов.

Особую роль в классе Beamer играют колонтитулы. Они не только отображают информацию о текущем разделе, но и оформляют ее в качестве гиперссылок. По умолчанию оформление колонтитулов на всех слайдах одинаковое.

При указании в параметрах ключевого слова `plain` на слайде отключаются колонтитулы, заголовки и отступы. Это может потребоваться при отображении больших изображений.

Титульный слайд

Для отображения титульной страницы используют команду **\titlepage**. Следующие два способа эквивалентны:

```
% Способ 1 (короткий)
\frame{\titlepage}
```

```
% Способ 2 (обычный)
\begin{frame}
  \titlepage
\end{frame}
```

Для сборки титула потребуется указать метаданные: название, авторов, дату и др. Их пишут в преамбуле документа, допускается комбинация различных вариантов.

- **\title** и **\subtitle** – заголовок и подзаголовок презентации.
- **\author** – автор(ы).
- **\date** – дата публикации.
- **\institute** – организация.
- **\logo** – логотип.

Пример

```
\documentclass{beamer}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{graphicx}

% Метаданные оглавления
\title{Верстка презентаций в \LaTeX}
```

```

\subtitle{Введение}
\author{Якубович Д.А., Еропова Е.С.}
\date{10 сентября 2019}
\institute{Владимирский государственный университет
имени\ \ Александра Григорьевича и Николая Григорьевича
Столетовых\ \ ВлГУ}
\logo{\includegraphics[height=1.0cm]{vlsu_logo.png}}

\begin{document}
  \frame{\titlepage}
\end{document}

```

Результат

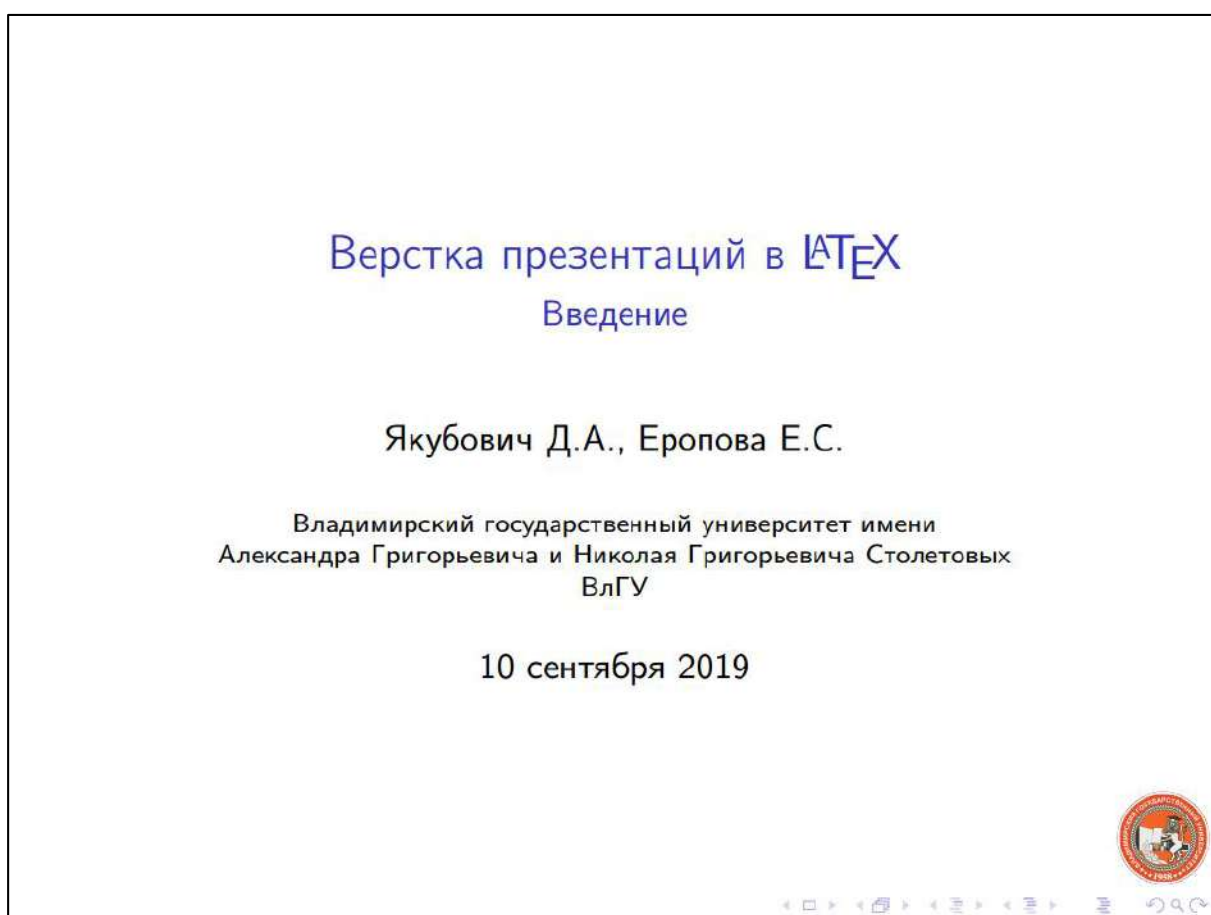


Рис. 3.73. Оформление титульного слайда.

Дополнительные опции помогут настроить более подробное описание слайда.

Пример

```
\documentclass{beamer}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\usepackage{graphicx}

\usetheme{Madrid} % Тема

% Метаданные оглавления
\title[Презентации в \LaTeX]{Верстка презентаций в
\LaTeX}
\subtitle[Лекция 1]{Введение}
\author[Якубович, Еропова]{Якубович Д.А.\inst{1} \and
Еропова Е.С.\inst{2}}
\date{10 сентября 2019}
\institute[ВлГУ]
{
  \inst{1}
  Владимирский государственный университет имени\\
  Александра Григорьевича и Николая Григорьевича
  Столетовых\\ ВлГУ
  \and
  \inst{2}
  Московский государственный университет\\
  МГУ
}
\logo{\includegraphics[height=1.0cm]{vlsu_logo.png}}

\begin{document}
  \frame{\titlepage}
\end{document}
```

Результат

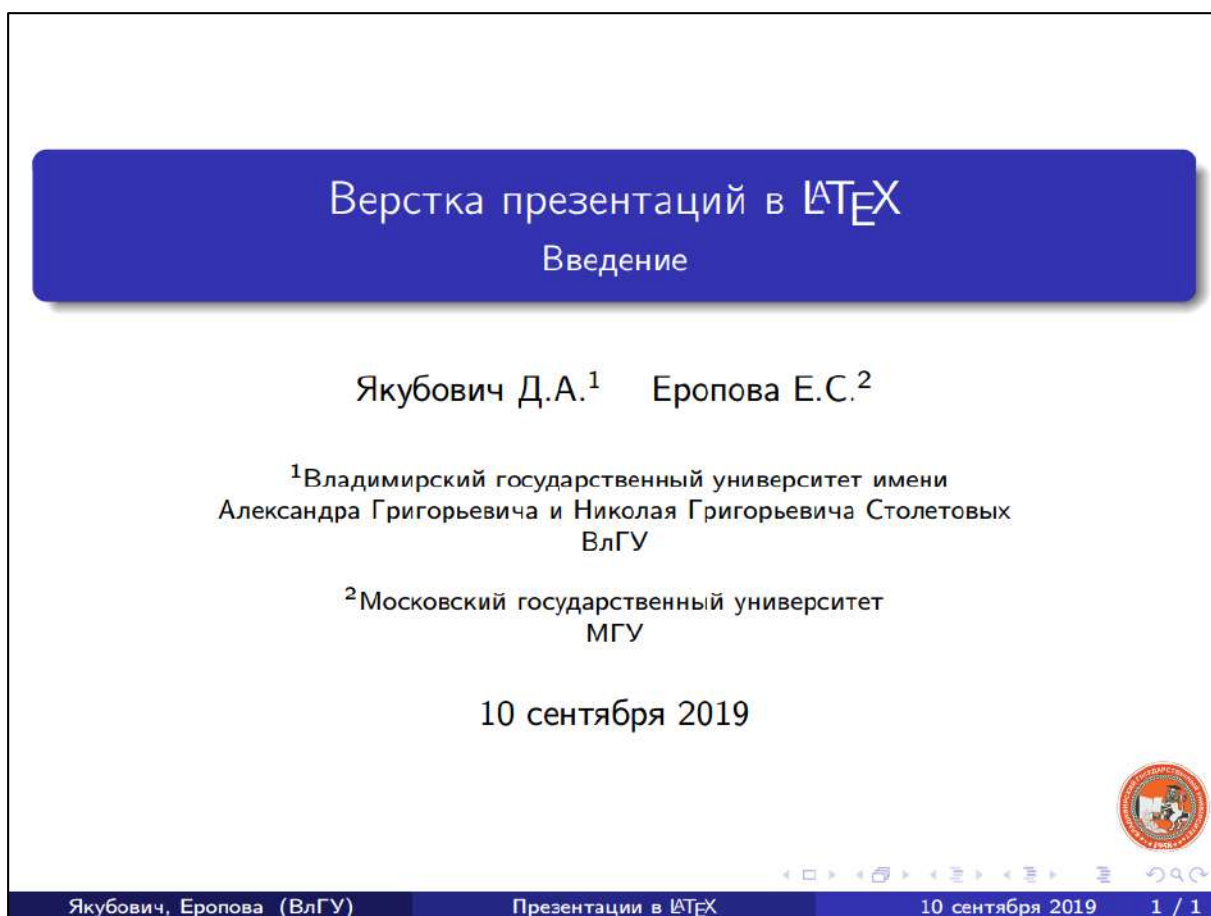


Рис. 3.74. Титульный слайд.

Тема оформления

Основные настройки внешнего вида презентации задаются выбранной темой. Beamer делит темы на несколько групп.

1. **Основная тема** – определяет внешний вид всех базовых элементов: параграфов, формул, боксов и т.д.
2. **Внутренняя тема** – формирует отображение элементов в рабочей области презентации (кроме колонтитулов, навигации и др.).
3. **Внешняя тема** – управляет форматированием внешних элементов (колонтитулов, полей и т.д.).
4. **Цветовая схема** – определяет цветовую гамму презентации. Допускается указание цветовых настроек отдельным элементам.

5. Тема шрифтов – задаёт используемые шрифты и их атрибуты.

Как правило, глубоких изменений базовых настроек не потребуется: стандартные схемы качественно настроены изначально.

Познакомиться со стандартными схемами можно на сайте http://deic.uab.es/~iblanes/beamer_gallery/.

AnnArbor



Antibes



Bergen



Рис. 3.75. Схемы оформления презентаций.

```
\usetheme[опции]{тема}
```

Задаёт тему оформления презентации.

Разные темы предполагают и разное содержимое, например, наличие или отсутствие навигации.

Пример

```
% Тема 1
\usetheme{Copenhagen}
```

```
% Тема 2
\usetheme{Hannover}
```

Результат

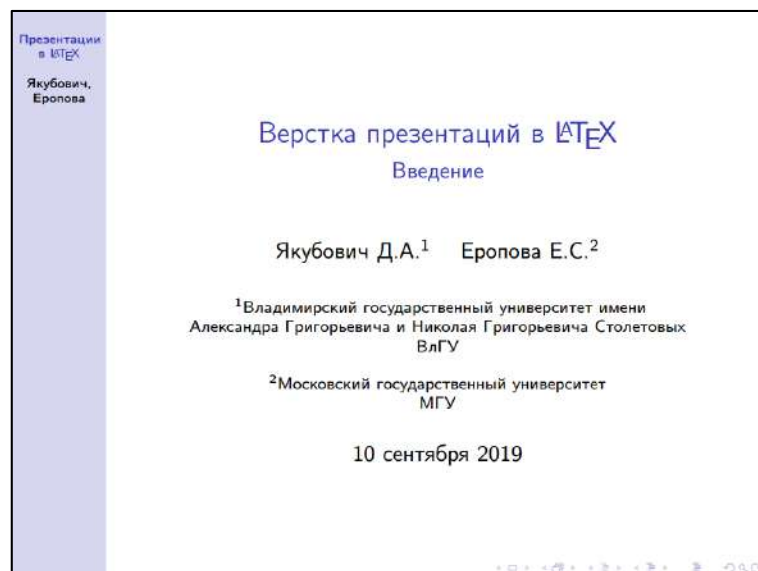
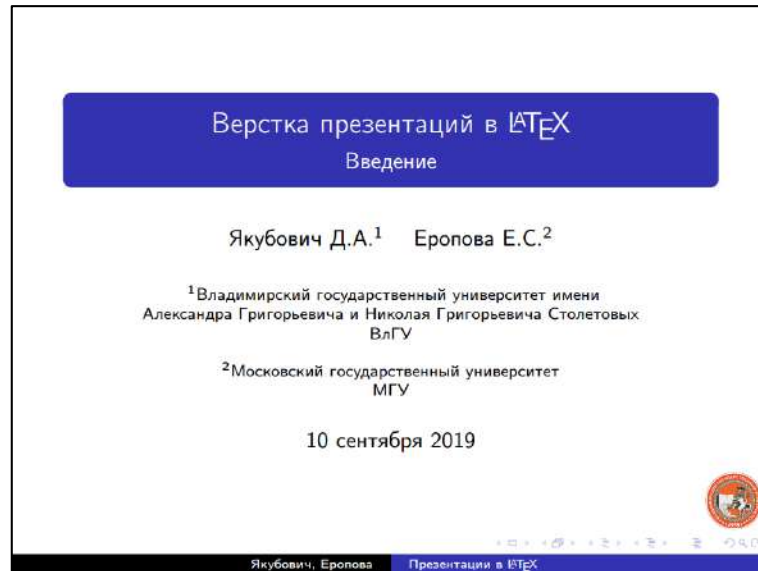


Рис. 3.76. Смена основной темы оформления презентации.

Настройку внешней темы можно осуществить командой `\useoutertheme`.

Пример

```
% Тема 1
\usetheme{Copenhagen}           % Тема
\useoutertheme{infolines}       % Внешняя тема
```

% Тема 2

`\usetheme{Hannover}`

% Тема

`\useoutertheme{miniframes}` % Внешняя тема

Результат

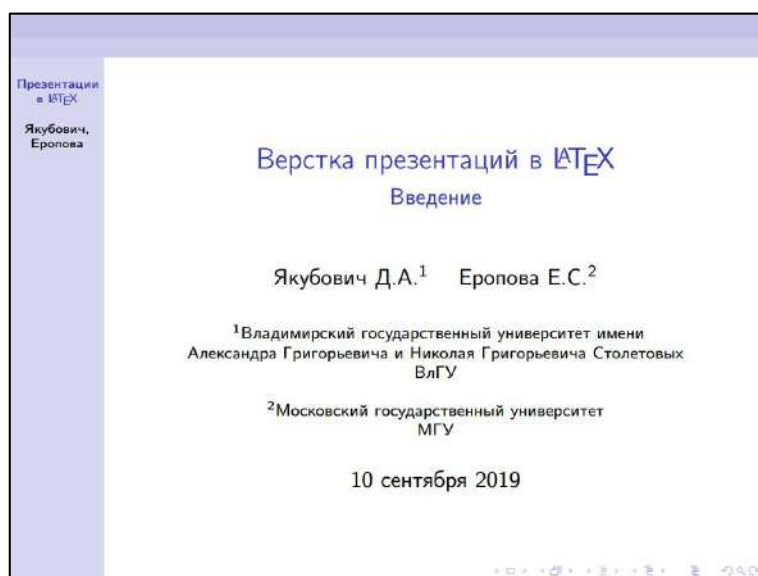
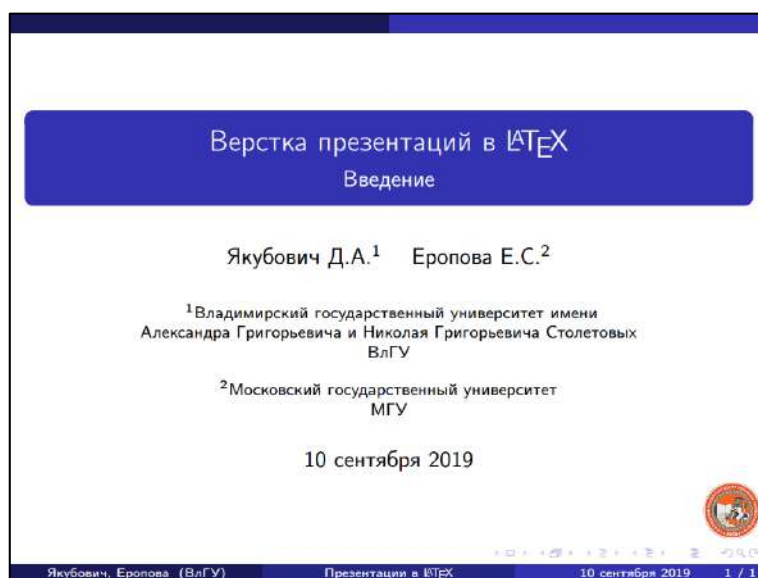


Рис. 3.77. Смена внешней темы оформления (появляется область для колонтитулов).

Для настройки цветовой схемы и особенностей гарнитуры шрифтов используются команды `\usecolortheme` и `\usefonttheme`:

Пример

`\usetheme{CambridgeUS}`

% Тема


```
\useoutertheme{miniframes} % Внешняя тема
\usecolortheme{crane}       % Цветовая схема
\usefonttheme[stillssansseriflarge]{serif} % Шрифт
```

Результат



Рис. 3.78. Изменение цветовой схемы и шрифта.

Графика и мультимедиа

Для включения изображений на слайд можно воспользоваться пакетом **graphicx** либо **pgf**. Оба пакета аналогичны, однако первый экономнее расходует память, а второй предоставляет возможность работы с полупрозрачными изображениями. В PDF-документ могут включаться лишь файлы с форматом .pdf, .jpg, .jpeg или .png.

Для пакета **graphicx** изображение подключается командой

```
\includegraphics[опции]{имя_файла}
```

Для пакета **pgf** это можно сделать командами

```
\pgfuseimage{имя_файла}
\pgfimage{имя_файла}
```

Пример

```
\begin{frame}[t]{Корень из единицы}
  Например, система корней 5-й степени из единицы
   $\sqrt[5]{1}$ :

  \begin{figure}[H]
    \center\includegraphics[width=0.7\linewidth]
    {cmplx_4.png}
  \end{figure}
\end{frame}
```

Результат

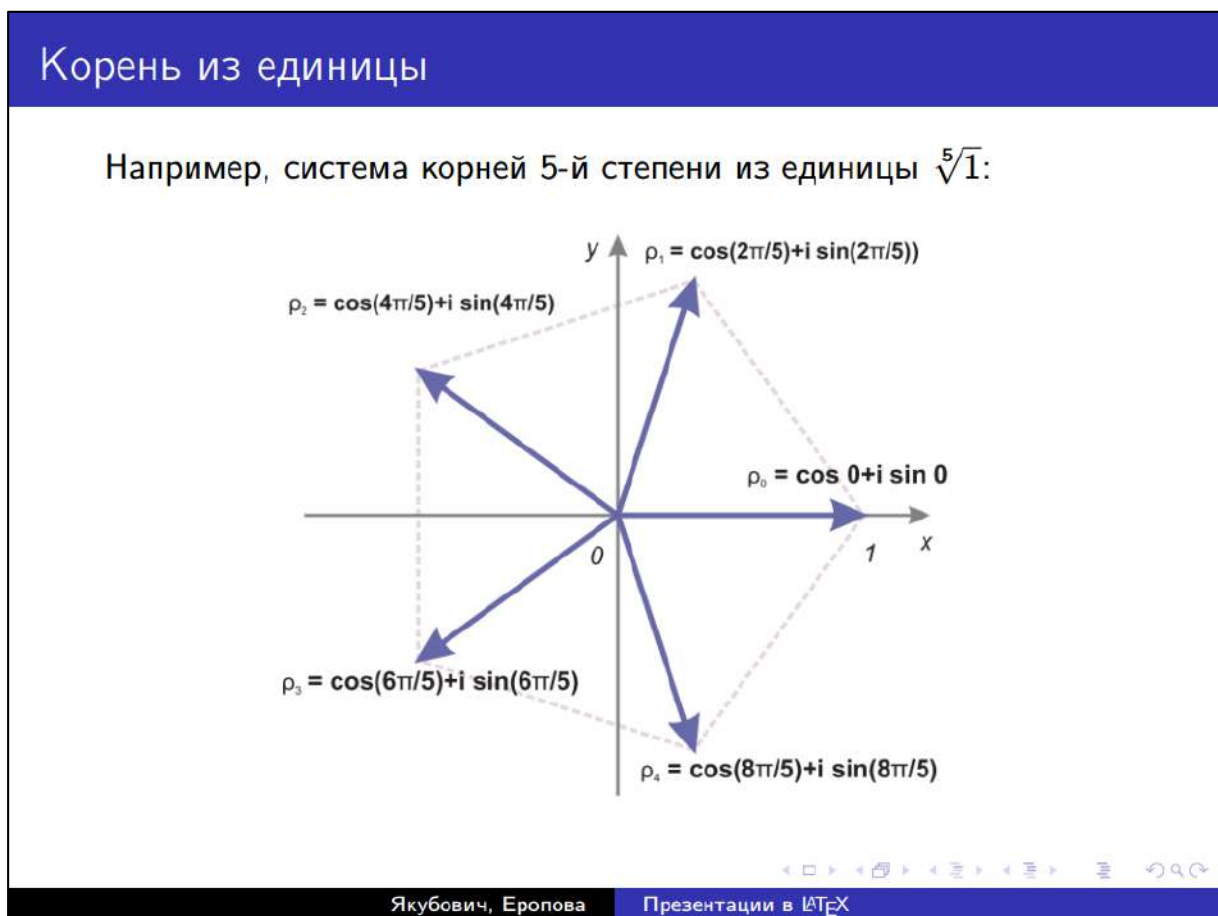


Рис. 3.79. Вставка изображения в презентацию.

Для работы с видеофайлами можно воспользоваться пакетами **multimedia**, **media9** и **movie15**.

Переход между соседними слайдами можно анимировать, добавив некоторые эффекты. Самый простой – команда `\pause`, ожидающая переход на следующий слайд (текущего фрейма). В PDF-документе создается «раскадровка» этого процесса: в полноэкранном

режиме просмотра при переходе на следующий слайд будет казаться, что текст последовательно появляется.

Пример

```
\begin{frame}
  \pause Первый слайд

  \pause Второй слайд

  \pause Третий слайд
\end{frame}
```

Результат

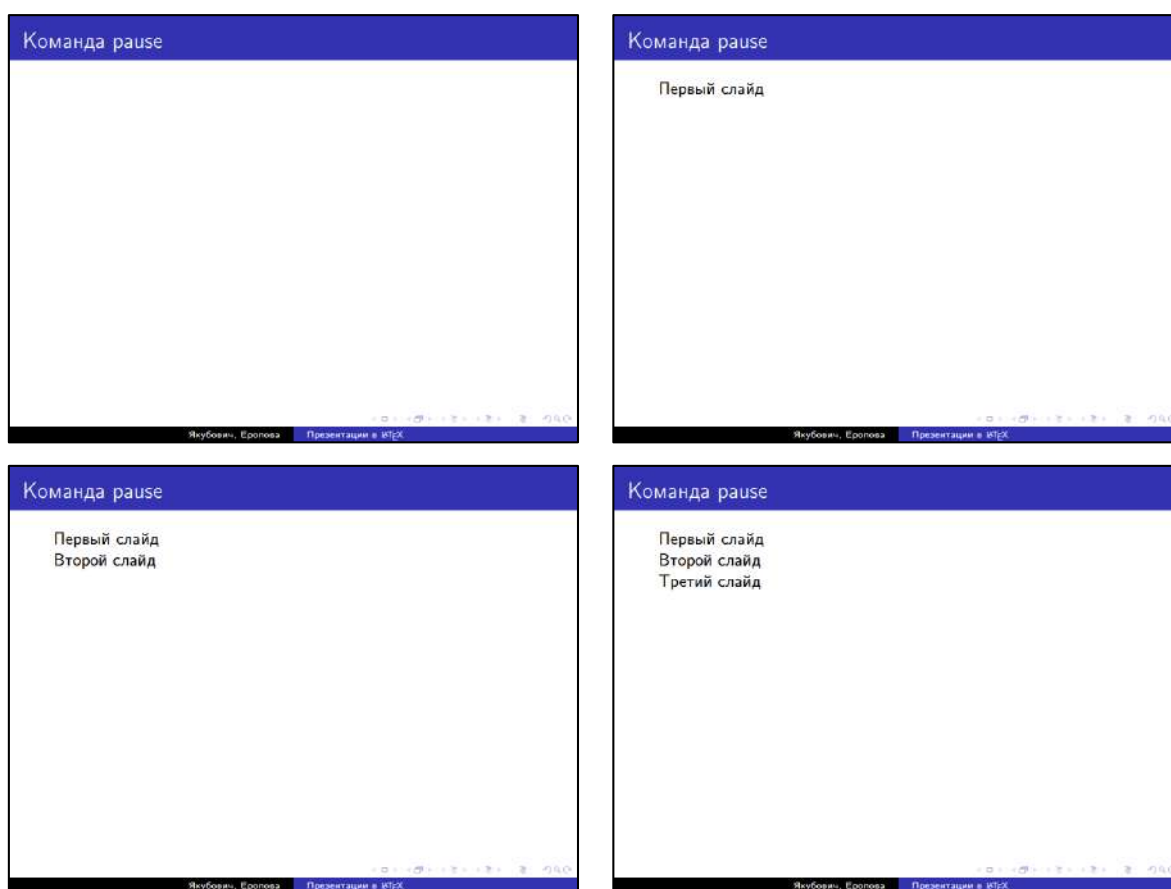


Рис. 3.80. Команда `\pause` разбивает фрейм на 4 подряд идущих слайда.

Гиперссылки и кнопки

За работу с гиперссылками отвечает уже известный читателю пакет **hyperref**. Ссылки могут быть определены на элементы текущего документа, внешние документы и ресурсы.

Для перехода к элементу внутри текущего документа этот элемент должен иметь уникальную метку, создаваемую командами `\hypertarget` или `\label`.

Для перехода к слайду, помеченному командами `\hypertarget` или `\label`, используется команда

```
\hyperlink<слайды>{метка}{текст}
```

Для вставки кнопок используйте команду

```
\beamerbutton{надпись}
```

Чтобы кнопка заработала, ее нужно поместить в гиперссылку.

В следующем примере создается фрейм, генерирующий поочередно элементы списка (для этого у каждого элемента списка указана команда `<+>`). На последнем слое появляется кнопка, по нажатию на которую пользователь возвращается на первый слайд.

Пример

```
\frame{\titlepage}
```

```
\begin{frame}[t]{Гиперссылки}
  \label{start}
```

```
  \begin{enumerate}
    \item<+> Пункт 1
    \item<+> Пункт 2
    \item<+> Пункт 3
  \end{enumerate}
```

```
  \hyperlink<3>{start<2>}{\beamerbutton{В начало}}
\end{frame}
```

Результат

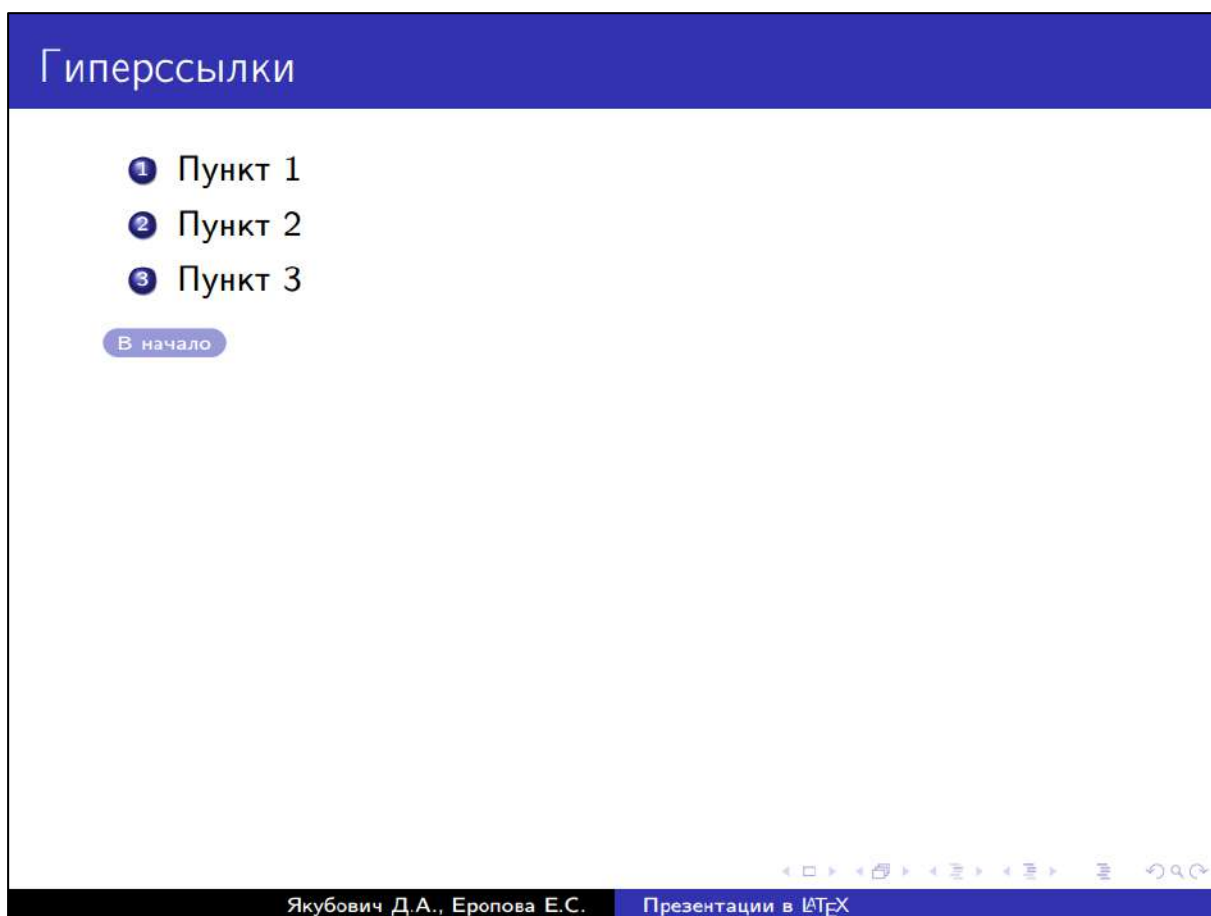


Рис. 3.81. Использование гиперссылок.

Рубрикация

Beamer поддерживает рубрикацию документа презентации. В зависимости от выбранной темы и ее настройки, она отображается в колонтитуле и, кроме того, каждый пункт является гиперссылкой на соответствующий подраздел.

Пример

```
\frame{\titlepage}

\section{Раздел}
\subsection{Подраздел 1}

\begin{frame}[t]{Фрейм 1}
    Содержимое фрейма 1
\end{frame}
```

```
\begin{frame}[t]{Фрейм 2}
  Содержимое фрейма 2
\end{frame}
```

```
\subsection{Подраздел 2}
```

```
\begin{frame}[t]{Фрейм 1}
  Содержимое фрейма 1
\end{frame}
```

```
\begin{frame}[t]{Фрейм 2}
  Содержимое фрейма 2
\end{frame}
```

```
\begin{frame}[t]{Фрейм 3}
  Содержимое фрейма 3
\end{frame}
```

Результат (на примере одного из слайдов)

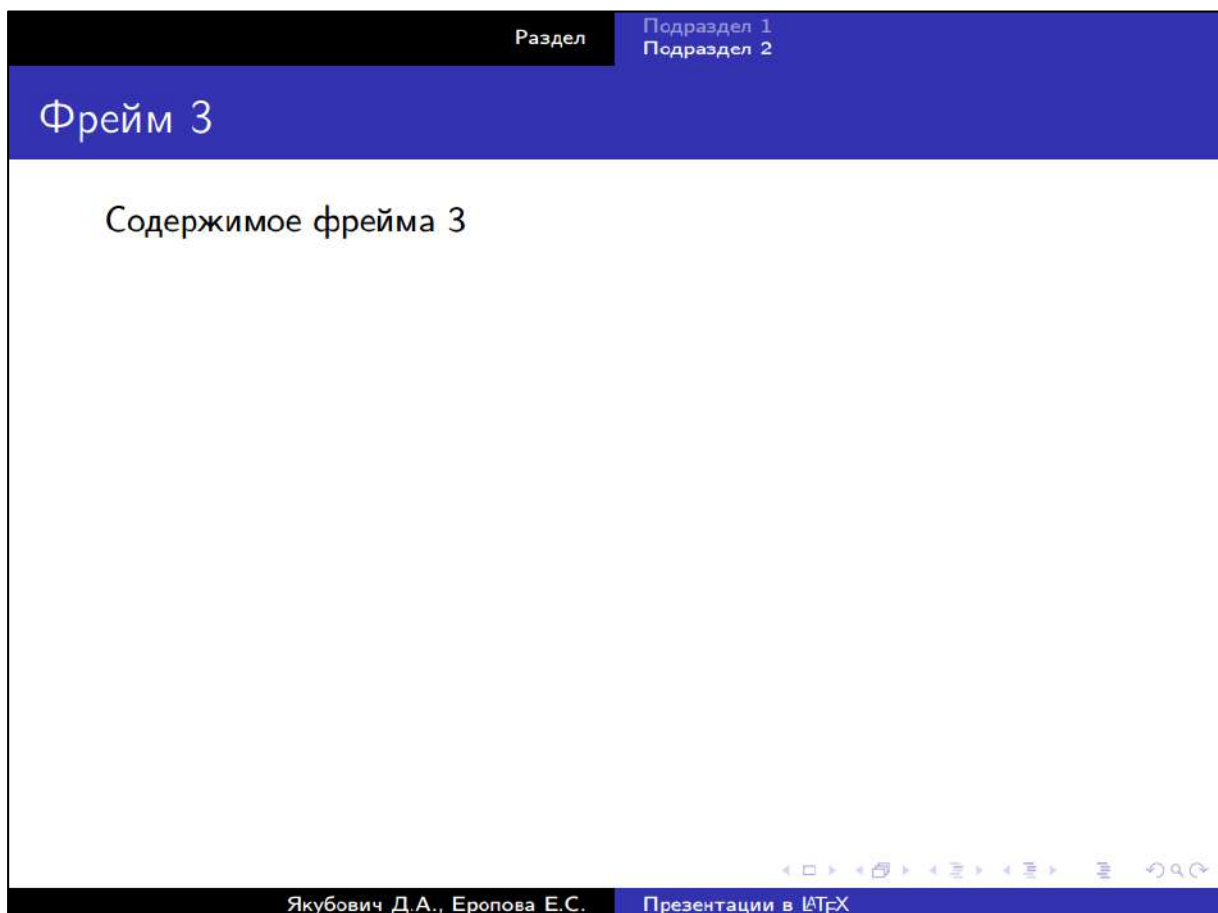


Рис. 3.82. Гипертекстовая рубрикация в презентации.

Вопросы для самопроверки

1. Перечислите основные рекомендации по оформлению презентаций.
2. Опишите возможности класса slides и его недостатки.
3. Перечислите возможности класса beamer.
4. Какие данные можно оформить на титуле презентации с помощью класса beamer?
5. Что значит тема презентации и где ее найти?
6. Допустима ли организация эффектов анимации в beamer?
7. Как организовать переход на определенный слайд?

Практикум

Форма отчета

1. Подготовить файлы заданий в форматах TEX и PDF.
2. Демонстрация задания преподавателю напрямую или по электронной почте.

1. Стандартный класс slides

На основе текста из занятия 3.6 составить слайды презентации, как изображено на рис. 3.83-рис. 3.85.

Для страницы установить следующие поля: слева и справа по 1.5 см, сверху и снизу – по 2 см.

2. Класс Beamer

Создайте титульный слайд (см. рис. 3.86), используя класс Beamer. В качестве темы оформления выберете Warsaw.

В разметке после команд вставки титульного слайда укажите разметку для разделов презентации и слайда с планом (оглавлением):

```
% слайд с оглавлением
\begin{frame}[t]{План}
  \tableofcontents
\end{frame}

\section{Теория}
\subsection{Определение}
\subsection{Дискриминант}
\subsection{Корни уравнения}
```

Заметим, что на этом этапе слайд с планом еще не будет содержать какой-либо информации (в силу отсутствия слайдов в других разделах).

Реализуйте три слайда первого подраздела презентации (рис. 3.87-рис. 3.89), стараясь сохранить форматирование текста и оформление элементов. Для вставки изображения используйте подход, описанный на занятии 3.3. Для того, чтобы отображались номера изображений, в преамбулу укажите команду

```
\setbeamertemplate{caption}[numbered]{} 
```

По аналогии, создайте слайды для второго и третьего подразделов (рис. 3.90-рис. 3.93).

В преамбулу документа вставьте команду

```
\usefonttheme[onlymath]{serif} 
```

Данная инструкция должна оформить шрифт для формул в классическом стиле.

Вернитесь на слайд с планом. Корректно ли он оформлен?

Квадратное уравнение

Якубович Д.А., Еропова Е.С.

1

1. Определение

Квадратное уравнение — это уравнение вида

$$ax^2 + bx + c = 0, \quad (1)$$

где коэффициенты a , b и c — произвольные числа, причем $a \neq 0$.

2

Рис. 3.83. Ожидаемый результат для задания 1 (слайды 1-2).

1. Определение

Классы квадратных уравнений

Все квадратные уравнения условно можно разбить на три класса:

1. имеют один корень;
2. имеют два различных корня;
3. не имеют корней.

3

2. Дискриминант

Дискриминантом называют число

$$D = b^2 - 4ac. \quad (2)$$

По знаку дискриминанта можно определить количество корней уравнения (1)

1. если $D < 0$, корней нет;
2. если $D = 0$, есть ровно один корень;
3. если $D > 0$, то уравнение (1) имеет два различных корня.

Например, следующее квадратное уравнение согласно формуле (2) имеет одно решение:

$$x^2 + 10x + 25 = 0.$$

4

Рис. 3.84. Ожидаемый результат для задания 1 (слайды 3-4).

3. Вычисление корней

$$D > 0$$

$$x_1 = \frac{-b - \sqrt{D}}{2a}, \quad x_2 = \frac{-b + \sqrt{D}}{2a}. \quad (3)$$

5

3. Вычисление корней

$$D = 0$$

$$x = -\frac{b}{2a}. \quad (4)$$

(4) является частным случаем (3).

6

Рис. 3.85. Ожидаемый результат для задания 1 (слайды 5-6).

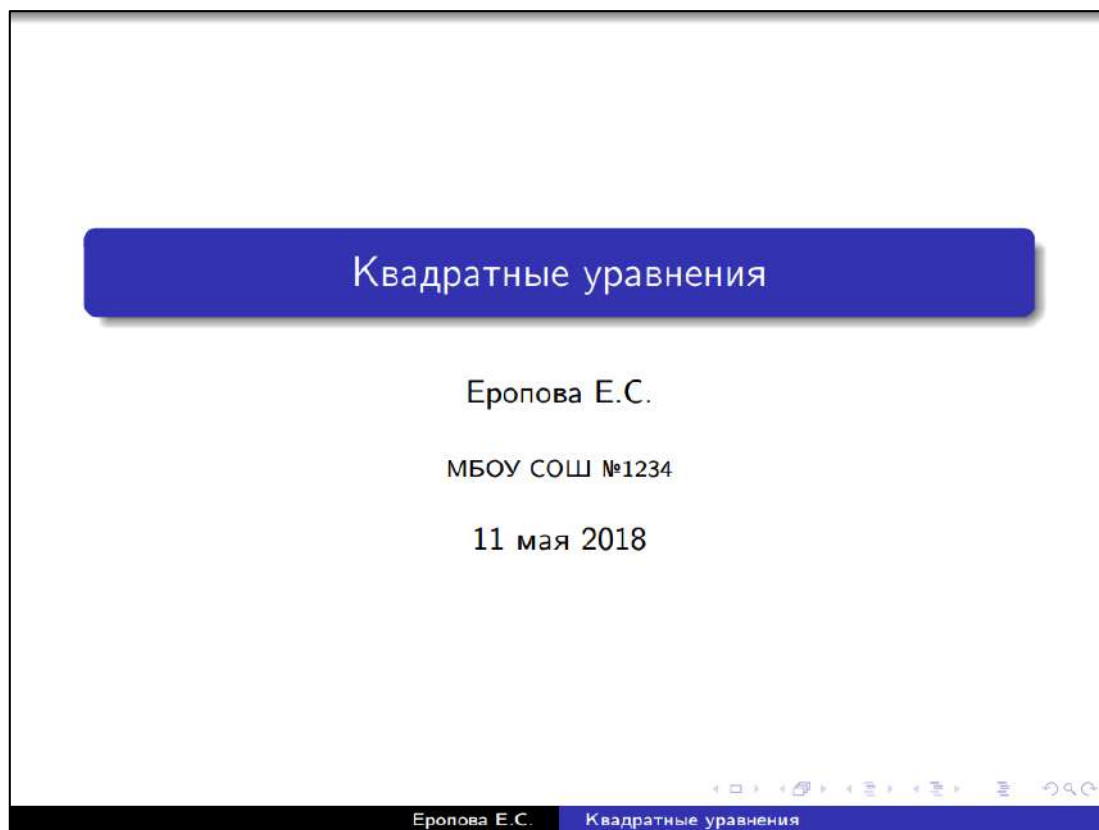


Рис. 3.86. Ожидаемый результат для задания 2 (титул).

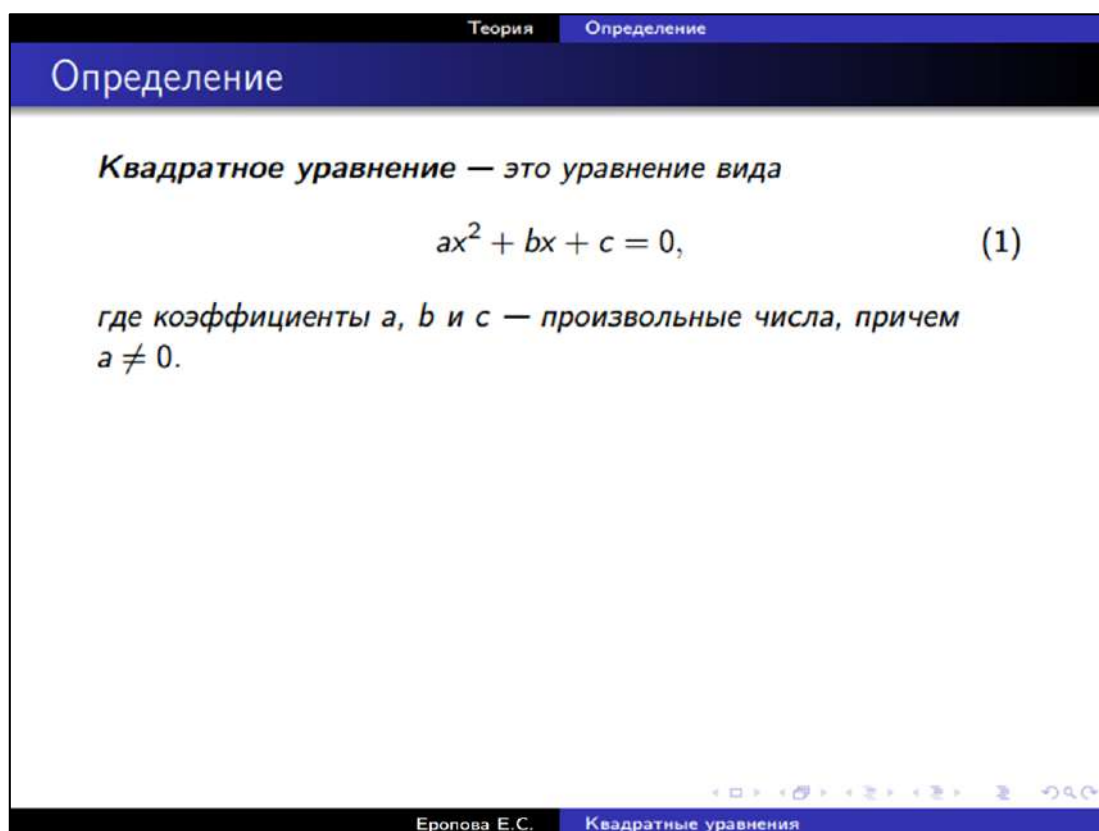


Рис. 3.87. Ожидаемый результат для задания 2 (первый раздел).

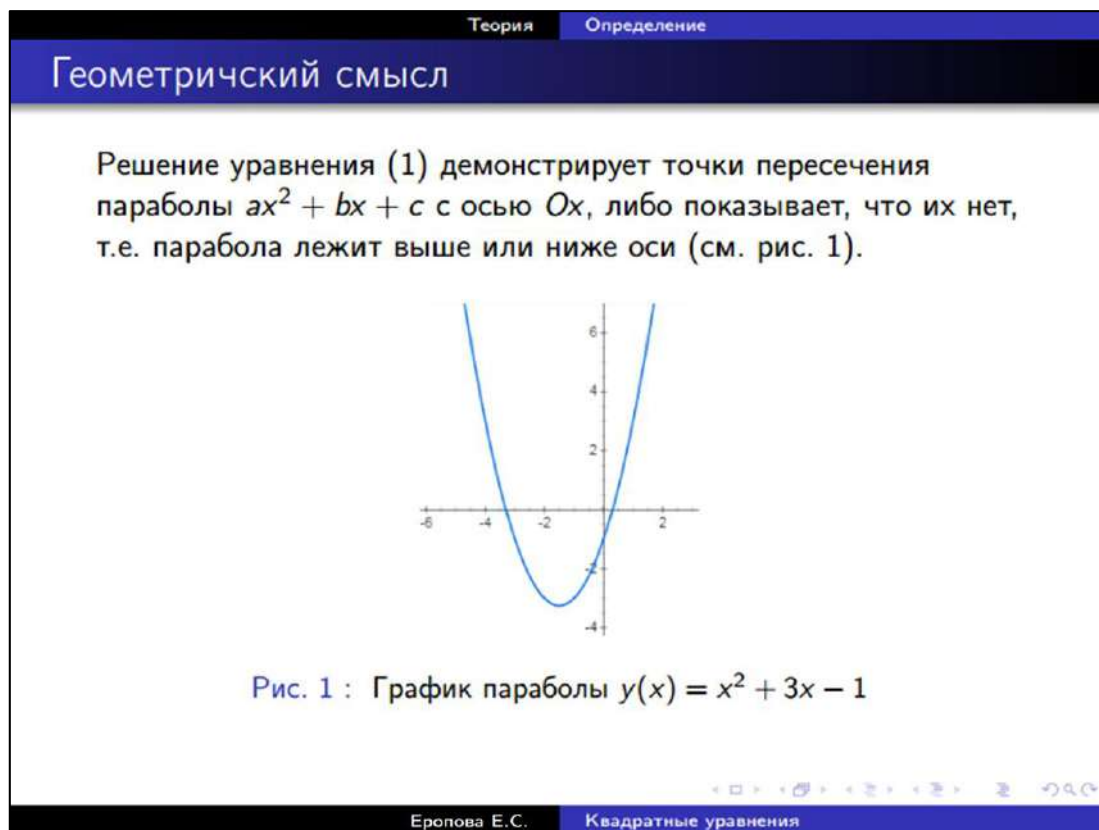


Рис. 3.88. Ожидаемый результат для задания 2 (первый раздел).

Теория Определение

Классы квадратных уравнений

Все квадратные уравнения условно можно разбить на три класса:

- ① имеют один корень;
- ② имеют два различных корня;
- ③ не имеют корней.

Еропова Е.С. Квадратные уравнения

Рис. 3.89. Ожидаемый результат для задания 2 (первый раздел).

Теория	Определение Дискриминант Корни уравнения
--------	--

Дискриминант

Дискриминантом называют число

$$D = b^2 - 4ac. \quad (2)$$

По знаку дискриминанта можно определить количество корней уравнения (1)

- ❶ если $D < 0$, корней нет;
- ❷ если $D = 0$, есть ровно один корень;
- ❸ если $D > 0$, то уравнение (1) имеет два различных корня.

⏪ ⏩ ⏴ ⏵ ⏶ ⏷ ⏸ ⏹ ⏺ ⏻ ⏼ ⏽ ⏾ ⏿ 🔍

Еропова Е.С. Квадратные уравнения

Рис. 3.90. Ожидаемый результат для задания 2 (второй раздел).

Теория	Определение Дискриминант Корни уравнения
--------	--

Примеры вычисления дискриминанта

Пример 1. Следующее квадратное уравнение согласно формуле (2) имеет одно решение:

$$x^2 + 10x + 25 = 0.$$

Пример 2. Указанное уравнение не имеет корней:

$$2x^2 + x + 5 = 0.$$

⏪ ⏩ ⏴ ⏵ ⏶ ⏷ ⏸ ⏹ ⏺ ⏻ ⏼ ⏽ ⏾ ⏿ 🔍

Еропова Е.С. Квадратные уравнения

Рис. 3.91. Ожидаемый результат для задания 2 (второй раздел).

Теория	Определение Дискриминант Корни уравнения
Вычисление корней	
<p>Если $D > 0$, то справедливо:</p> $x_1 = \frac{-b - \sqrt{D}}{2a}, \quad x_2 = \frac{-b + \sqrt{D}}{2a}. \quad (3)$	
<p style="text-align: right;"> </p>	
Еропова Е.С.	Квадратные уравнения

Рис. 3.92. Ожидаемый результат для задания 2 (третий раздел).

Теория	Определение Дискриминант Корни уравнения
Вычисление корней	
<p>Если $D = 0$, то справедливо:</p> $x = -\frac{b}{2a}. \quad (4)$ <p>(4) является частным случаем (3).</p>	
<p style="text-align: right;"> </p>	
Еропова Е.С.	Квадратные уравнения

Рис. 3.93. Ожидаемый результат для задания 2 (третий раздел).

3.9 Примеры шаблонов документов

3.9.1 Шаблон делового письма

Ведение деловой переписки является неотъемлемой частью работы специалиста сферы образования и науки.

Как правило формат делового письма не ограничивается жесткими рамками. Однако, например, у нас и за рубежом в научных кругах сформировались определенные правила по их оформлению.

При написании делового письма следует обеспечить наличие следующих критериев:

- четкая структура письма;
- соблюдение интернациональных (региональных) правила приветствия и прощания;
- текст письма – не более одной страницы;
- присутствие данных об адресате, дате, теме письма;
- отделение абзацев пустой строкой.

Шаблон

```
% Подключаем класс по оформлению писем
\documentclass{letter}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

%\usepackage[top=2.5cm, bottom=3cm, left=3cm,
right=1.5cm]{geometry}
\usepackage{hyperref}      % Использование гиперссылок

\linespread{1.15}
\parindent = 0.75cm

% Ключевые данные
\signature{Якубович~Д.А.}
\address{Россия\\ Владимир \\ пр. Строителей, д. 11\\
Тел. 11-22-33 \\ mail@mail.ru }
\date{\today}

\begin{document}
```



```

\begin{letter}{Заведующему кафедрой\\
математического образования и информационных
технологий \\ВлГУ}
  % Слова приветствия
  \opening{Уважаемая Юлия Юрьевна!}

  % Текст письма
  Обращаюсь к Вам по вопросу организации ряда
  пробных занятий, посвященных теме моей диссертации.
  Прошу \ldots

  \ldots

  Благодарю Вас за внимание к этому письму.

  В ожидании Вашего ответа,

  \closing{студент кафедры МОиИТ, Якубович~Д.А.}

  \ps{P.S. В приложении к данному письму, высылаю
  Вам ... } % P.S.

  Контактные данные:
  \href{mailto:mail@mail.ru}{mail@mail.ru}
\end{letter}

\end{document}

```

Результат изображен на рис. 3.94.

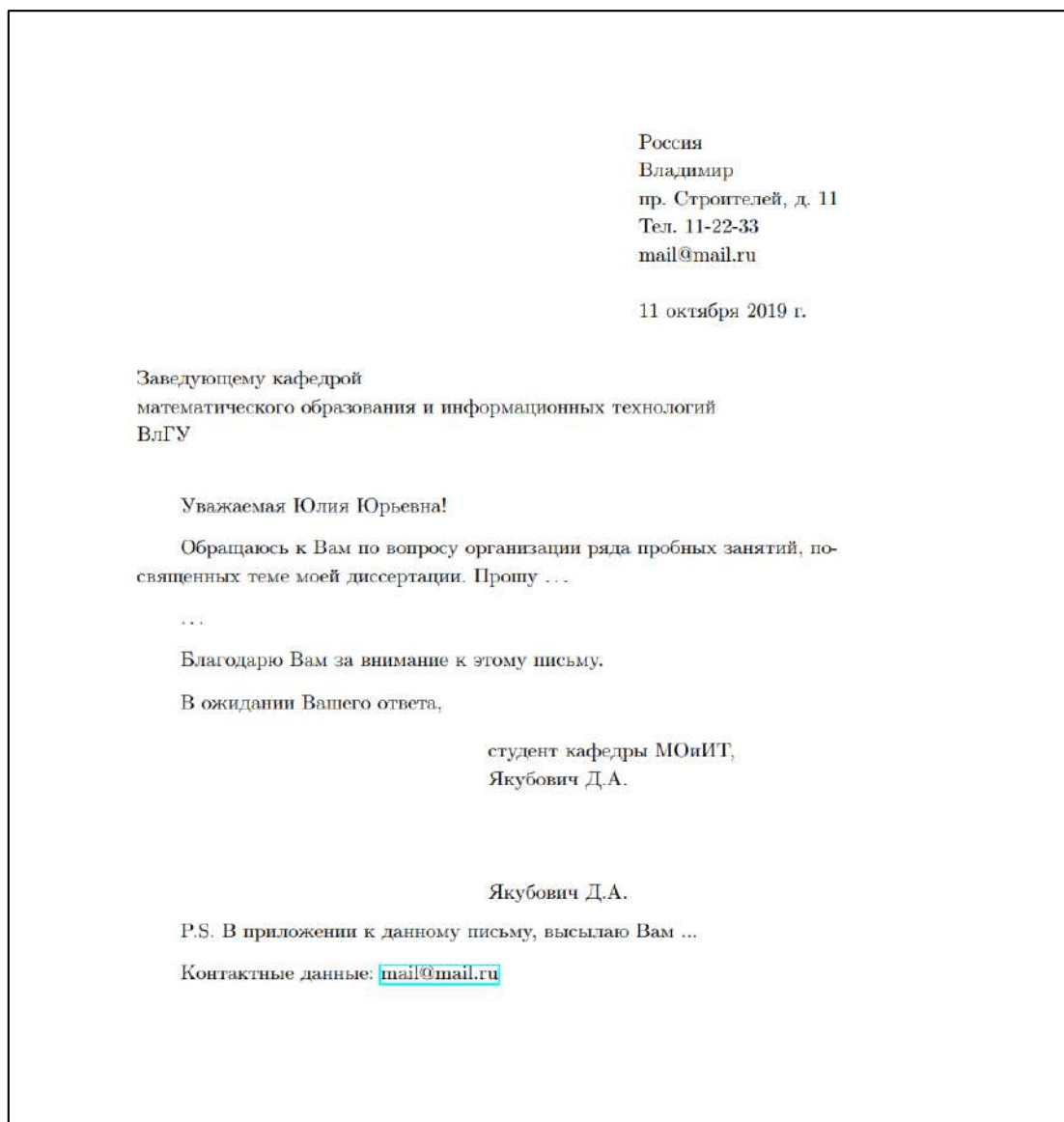


Рис. 3.94. Пример оформления письма.

Ссылки на Интернет-ресурсы

Ниже представлены ссылки на ресурсы с примерами шаблонов писем:

- Сайт Wikibooks.org. Адрес сайта:
<https://en.wikibooks.org/wiki/LaTeX/Letters>.
- Шаблон официального письма LaTeX. Сайт:
<http://gorod1277.org/?q=content/shablon-ofitsialnogo-pisma-latex>.
- Шаблоны писем, ShareLaTeX. Сайт:
<https://ru.sharelatex.com/templates/cover-letters>.

- Шаблоны писем, Overleaf.com. Сайт:
<https://www.overleaf.com/latex/templates?addsearch=letter>.

3.9.2 Шаблоны научных статей

В отличие от делового письма, научная статья требует четкого соблюдения стандартов по ее оформлению. Последние могут определяться:

- требованиями научного издания;
- учебным заведением (если работа публикуется в журнале вуза);
- другими стандартами, в частности, ГОСТ (оформление ВКР, рефератов, диссертаций, др.).

Шаблоны

Приведем пример шаблона научной статьи для журнала естественно-научного цикла. В зависимости от требований, он может быть упрощен, расширен и даже существенно модифицирован. Тем не менее, в шаблоне читатель найдет наиболее часто используемые разделы любой статьи.

В разметке приведены комментарии. Рекомендуем внимательно изучить код разметки и создать документ самостоятельно.

```
% Научная статья
\documentclass[12pt,a4paper]{article}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english, russian]{babel}

% Отступ красной строки у первого абзаца после заголовка
% (по умолчанию не ставится)
\usepackage{indentfirst}

% Пакеты серии AMS (для оформления формул)
\usepackage{amsmath,amssymb,amsthm,amsfonts,amscd}

% Пакет ставит в номерах раздела точки
% (не всегда требуется, см. ГОСТ)
\usepackage{secdot}

\usepackage{graphicx}      % Вставка изображений
\usepackage{color}         % Работа с цветовыми моделями
```

```

\usepackage{geometry}      % Геометрия полей страницы
\geometry{left=3cm}
\geometry{right=3cm}
\geometry{top=3cm}
\geometry{bottom=3cm}

% Создаем окружения на базе AMS-стилей для оформления
% теорем, определений, свойств и т.д.
% Последние автоматически оформляются и нумеруются в
% рамках секций (для каждого будет создан свой счетчик)
\newtheorem{definition}{Определение}[section]
\newtheorem{theorem}{Теорема}[section]
\newtheorem{lemma}{Лемма}[section]
\newtheorem{conseq}{Следствие}[section]
\newtheorem{comm}{Замечание}[section]

% Формулы нумеруются не подряд,
% а внутри секций (\section)
\numberwithin{equation}{section}

% В оглавлении выставляем точки-заполнители
\usepackage{tocloft}
\renewcommand{\cftsecleader}{\cftdotfill{\cftdotsep}}

% Описываем метаданные
\title{Разработка учебного курса по использованию
издательской системы LaTeX в работе современного учителя}

% Окружение minipage позволяет создавать колонки
% В данном случае данные об авторах разбиты на две
% колонки. Ширина каждой - 45% от доступной ширины
% текстовой области. 10% остается на отступы между
% колонками и справа / слева.
% Сноски указывают дополнительную информацию
\author{
  \small
  \begin{minipage}[t]{0.45\textwidth}
    \begin{center}
      \textbf{Якубович Д.А.}\footnote{Работа
        поддержана Министерством добрых дел
        (XXX - XXX - XX).}
    \end{center}
  \end{minipage}
  \begin{minipage}[t]{0.45\textwidth}
    \begin{center}
      \textbf{Владимирский государственный университет
        имени А.Г. и Н.Г. Столетовых}
    \end{center}
  \end{minipage}
}

```

```

        \\
        \textit{yakubovichfmf@mail.ru}
    \end{center}
\end{minipage}
\begin{minipage}[t]{0.45\textwidth}
    \begin{center}
        \textbf{Еропова Е.С.}\footnote{Работа
            Поддержана Министерством добрых дел
            (YYY - YY - YY).
        }
    \end{center}
    \\
    Владимирский государственный университет
    имени А.Г. и Н.Г. Столетовых
    \\
    \textit{eropova13061962@mail.ru}
    \end{center}
\end{minipage}
}

\date{}    % Дату не указываем

\begin{document}
    \maketitle          % Собираем информацию титула
    \tableofcontents    % Генерируем оглавление

    \vspace{1em}

    \begin{abstract}    % Аннотация к статье
        Текст аннотации.
    \end{abstract}

    \linespread{1.05}   % Междустрочный интервал
    \parindent=1.0cm    % Отступ красной строки

    % Рубрикация
    \section{Введение}
    Текст введения. . .

    Отметим популярные источники по \LaTeX в
    русскоязычном сегменте:
    \cite{Lvovsky}, \cite{KotCheb}.

    \section{Пункт работы 2}
    Следующим образом оформляется утверждение:

```

```

\begin{theorem}\label{thrm:pif}
  В прямоугольном треугольнике квадрат гипотенузы
  равен сумме квадратов катетов:
  \begin{equation}\label{eq:pif}
    c^2 = a^2 + b^2.
  \end{equation}
\end{theorem}

```

Согласно теореме~\ref{thrm:pif} и
 формуле~\eqref{eq:pif}
 . . .

```

\begin{theorem}
  Теоремы автоматически нумеруются.
\end{theorem}

```

```

\begin{comm}
  Это окружение оформит замечание.
\end{comm}

```

```

\subsection{Подпункт работы 2.1}
Текст . . .

```

```

\subsection{Подпункт работы 2.1}
Текст . . .

```

```

\section{Пункт работы 3}
Текст . . .

```

```

\section{Пункт работы 4}
Текст . . .

```

```

% Без нумерации и поместим в оглавление
\section*{Заключение}
\addcontentsline{toc}{section}{Заключение}

```

```

% Переключаем в режим нумерации приложений
\appendix
\section{Приложение 1}
\subsection{Приложение 1.1}
\subsection{Приложение 1.2}
\subsection{Приложение 1.3}

```

```

% Включаем в оглавление

```

```

\addcontentsline{toc}{section}{Список литературы}
\begin{thebibliography}{99}

  \bibitem{Lvovsky}
  Львовский, С.М.
  Набор и верстка в системе LaTeX /
  С.М. Львовский. – М.: МЦНМО, 2014. - 398 с.

  \bibitem{KotCheb}
  Котельников, И. А., Чеботаев, П. З.
  LaTeX2e по-русски. –
  Новосибирск: Сибирский Хронограф, 2004. - 496 с.

\end{thebibliography}

\end{document}

```

Заготовка для статьи выглядит, как изображено на
рис. 3.95-рис. 3.97.

Разработка учебного курса по использованию издательской системы LaTeX в работе современного учителя

Якубович Д.А.*

Владимирский государственный
университет имени А.Г. и Н.Г.
Столетовых
yakubovichfmf@mail.ru

Еропова Е.С.†

Владимирский государственный
университет имени А.Г. и Н.Г.
Столетовых
eropova13061962@mail.ru

Содержание

1 Введение	1
2 Пункт работы 2	2
2.1 Подпункт работы 2.1	2
2.2 Подпункт работы 2.1	2
3 Пункт работы 3	2
4 Пункт работы 4	2
Заключение	3
A Приложение 1	3
A.1 Приложение 1.1	3
A.2 Приложение 1.2	3
A.3 Приложение 1.3	3
Список литературы	3

Аннотация

Текст аннотации.

1. Введение

Текст введения. . .

*Работа поддержана Министерством добрых дел (XXX - XXX - XX).

†Работа поддержана Министерством добрых дел (YYY - YYY - YY).

Рис. 3.95. Шаблон статьи (стр. 1).

Отметим популярные источники по L^AT_EX в русскоязычном сегменте: [1], [2].

2. Пункт работы 2

Следующим образом оформляется утверждение:

Теорема 2.1. *В прямоугольном треугольнике квадрат гипотенузы равен сумме квадратов катетов:*

$$c^2 = a^2 + b^2. \quad (2.1)$$

Согласно теореме 2.1 и формуле (2.1) . . .

Теорема 2.2. *Теоремы автоматически нумеруются.*

Замечание 2.1. *Это окружение оформит замечание.*

2.1 Подпункт работы 2.1

Текст . . .

2.2 Подпункт работы 2.1

Текст . . .

3. Пункт работы 3

Текст . . .

4. Пункт работы 4

Текст . . .

Рис. 3.96. Шаблон статьи (стр. 2).

Заключение

А. Приложение 1

А.1 Приложение 1.1

А.2 Приложение 1.2

А.3 Приложение 1.3

Список литературы

- [1] Львовский, С.М. Набор и верстка в системе LaTeX / С.М. Львовский. – М.: МЦНМО, 2014. - 398 с.
- [2] Котельников, И. А., Чеботаев, П. З. LaTeX2e по-русски. – Новосибирск: Сибирский Хронограф, 2004. - 496 с.

Рис. 3.97. Шаблон статьи (стр. 3).

Ссылки на Интернет-ресурсы

- Шаблоны научных статей, сайт Overleaf.com:
<https://www.overleaf.com/gallery/tagged/academic-journal>.
- Шаблоны научных статей, сайт ShareLaTeX:
<https://ru.sharelatex.com/templates/journals>.
- Шаблоны научных статей и журналов, сайт LaTeXTemplates.com:
<http://www.latextemplates.com/cat/articles>;
<http://www.latextemplates.com/cat/academic-journals>.

3.9.3 Шаблоны книг

Оформление крупных изданий (книг, монографий) требует скрупулёзной работы. Нужно учитывать множество деталей: от тонкостей оформления до корректности перекрестных ссылок и навигации между разделами.

Стандартный класс book поможет вам оформить конспект занятий, учебное пособие, а также любой тест большого объема.

Шаблоны

Для оформления стандартной книги будет достаточным взять за основу следующий шаблон с текстом и разметкой секций (для краткости текст из абзацев приводится не полностью). Разумеется, дополнительные разделы (например, библиографию), пакеты и настройки можно вносить по мере необходимости.

Код

```
\documentclass[12pt]{book}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

\title{\Huge\textbf{\textsc{Полиграфия}}}}
\author{ru.wikipedia.org}
\date{\today}

\linespread{1.45}
\parindent=1.0cm
```

```

\begin{document}
  \maketitle

  \tableofcontents

  \chapter[Основы полиграфии]{Основы полиграфии}
  \section*{Введение}
  \addcontentsline{toc}{section}{Введение}

  \textbf{Полиграфия} --- отрасль промышленности,
  занимающаяся ...

  \section{Основные типы печати}
  К основным типам печати относят ...

  \section{Особые виды полиграфии}
  \subsection{Ароматическая полиграфия}
  \textbf{Ароматическая полиграфия} --- полиграфия с
  применением ...

  \subsection{Стерео}
  \textbf{Стерео} --- ярко выраженный эффект ...

  \section{Возможные дефекты}
  \subsection{Отмарывание}
  \textbf{\textit{Отмарывание}} -- переход краски ...

  \subsection{Марашки}
  \textbf{\textit{Марашки}} -- белые пятна, ...
  \subsection{Разнооттеночность оттисков}
  \textbf{\textit{Разнооттеночность оттисков}} -- ...
  \subsection{Деформация растровых точек}
  \textbf{\textit{Деформация растровых}} -- ...

  \subsection{Двоение печатных элементов}
  \textbf{\textit{Двоение печатных элементов}} -- ...
  \subsection{Муар}
  \textbf{\textit{Муар}} -- ...

  \subsection{Полошение}
  \textbf{\textit{Полошение}} -- ...
  \section{Программы для автоматизации учета в
  типографиях}
  ...

```

```
\section{Перспективы полиграфической отрасли}
...
```

```
\end{document}
```

Важный элемент – навигация в PDF-документе. Следующий пример демонстрирует некоторые ее элементы.

Код

```
\documentclass[12pt]{book}
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}

% Пакет для работы с цветовыми моделями
\usepackage{xcolor}
% Пакет для работы с ссылками
\usepackage{hyperref}

% Настройка некоторых параметров ссылок
\hypersetup{
    colorlinks = true,           % Ссылки цветные? - Да
    linkcolor = {red!75!black}, % Внутренние ссылки -
                                % красно-черный (75%)
    urlcolor = blue,            % Цвет ссылок на внешние
                                % (Интернет) ресурсы
    bookmarks = true,          % Создавать закладки
                                % оглавления в PDF? - Да
}

\title{\Huge\textbf{\textsc{Полиграфия}}}}
\author{\href{https://ru.wikipedia.org/}
{ru.wikipedia.org}}
\date{\today}

\linespread{1.45}
\parindent=1.0cm

\begin{document}
    % По аналогии с предыдущим
\end{document}
```

Некоторые из страниц документа изображены на рис. 3.98-рис. 3.101.

ПОЛИГРАФИЯ

ru.wikipedia.org

7 мая 2018 г.

Рис. 3.98. Титульный лист.

Оглавление

1 Основы полиграфии	5
Введение	5
1.1 Основные типы печати	5
1.2 Особые виды полиграфии	6
1.2.1 Ароматическая полиграфия	6
1.2.2 Стерео	6
1.3 Возможные дефекты	7
1.3.1 Отмарывание	7
1.3.2 Марашки	7
1.3.3 Разнооттеночность оттисков	7
1.3.4 Деформация растровых точек	8
1.3.5 Двоение печатных элементов	8
1.3.6 Муар	8
1.3.7 Полошение	8
1.4 Программы для автоматизации учета в типографиях	8
1.5 Перспективы полиграфической отрасли	9

Рис. 3.99. Оглавление.

1.2 Особые виды полиграфии

1.2.1 Ароматическая полиграфия

Ароматическая полиграфия — полиграфия с применением ароматических красок и лаков. Одна из существующих технологий основана на добавлении ароматических масел в типографскую краску.

Но более популярны две схожие технологии, которые по аналогии с английскими названиями можно назвать «поскреби и понюхай» и «оторви и понюхай». Технологии основаны на добавлении ароматических веществ, заключенных в микрокапсулы в типографскую краску. В первом случае, для получения запаха, по зоне ароматической печати необходимо поскрести, чтобы разрушить оболочки микрокапсул. Во втором случае участок с ароматической печатью прикрывается наклеенной бумагой, при отрыве которой аналогичным образом разрушаются микрокапсулы и высвобождается запах.

1.2.2 Стерео

Стерео — ярко выраженный эффект объема композиции или объекта.

Этот эффект основан на том, что при просмотре полученного стереоизображения один глаз видит объект с одного ракурса, другой глаз — с другого. Таким образом, вы видите объект так, что возникает видимость глубины изображения. Для создания стереоэффекта необходима многоракурсная съёмка или 3D-моделирование. Одной из разновидностей стереоэффекта является т. н. «псевдостерео». Эффект основан на взаимном смещении слоёв композиции относительно друг друга и пере-

Рис. 3.100. Одна из страниц книги.

Оглавление

1 Основы полиграфии	5
Введение	5
1.1 Основные типы печати	5
1.2 Особые виды полиграфии	6
1.2.1 Ароматическая полиграфия	6
1.2.2 Стерео	6
1.3 Возможные дефекты	7
1.3.1 Отмарывание	7
1.3.2 Марашки	7
1.3.3 Разнооттеночность оттисков	7
1.3.4 Деформация растровых точек	8
1.3.5 Двоение печатных элементов	8
1.3.6 Муар	8
1.3.7 Полошение	8
1.4 Программы для автоматизации учета в типографиях	8
1.5 Перспективы полиграфической отрасли	9

Рис. 3.101. Оформление оглавления с гипертекстовыми ссылками.

Ссылки на Интернет-ресурсы

- Шаблоны книг, сайт Overleaf.com:
<https://www.overleaf.com/gallery/tagged/book>.
- Шаблоны книг, сайт ShareLaTeX:
<https://ru.sharelatex.com/templates/books>.
- Шаблоны книг, сайт LaTeXTemplates.com:
<http://www.latextemplates.com/cat/books>.

3.9.4 Шаблоны презентаций

Если вам требуется оформить качественную презентацию для официального мероприятия, конференции, лекции, то класс Beamer даст вам необходимое.

Шаблоны

Читатель может воспользоваться примерами, рассмотренными в рамках занятия 3.8. В этом параграфе мы попытались обозначить основные вопросы, касаемые разработки презентаций на базе класса Beamer.

Ссылки на Интернет-ресурсы

- Шаблоны презентаций, сайт Overleaf.com:
<https://www.overleaf.com/gallery/tagged/presentation>.
- Шаблоны презентаций, сайт ShareLaTeX:
<https://ru.sharelatex.com/templates/presentations>.
- Шаблоны презентаций, сайт LaTeXTemplates.com:
<http://www.latextemplates.com/cat/presentations>.
- Балдин Е. М., разработка презентаций (PDF-документ):
<https://star.inp.nsk.su/~baldin/LaTeX/lurs-presentation.pdf>.

3.9.5 Другие шаблоны

Помимо указанных типов документов, среди шаблонов можно найти и множество других. В первую очередь, такие ресурсы, как Overleaf.com, ShareLaTeX.com, предоставляют доступ к сотням интересных и полезных шаблонов самых разнообразных документов.

Работа со множествами шаблонов потребует дополнительного изучения новых команд и пакетов, которые мы не затрагивали в учебном курсе.

Ссылки на Интернет-ресурсы

- Шаблоны Overleaf.com:
<https://www.overleaf.com/latex/templates>.
- Шаблоны ShareLaTeX.com:
<https://ru.sharelatex.com/templates>.
- Шаблоны LaTeXTemplates.com:
<https://www.latextemplates.com/>.

Практикум

Итоговый тест

Проверьте себя по всему курсу LaTeX. В каждом вопросе корректным является только один ответ. Правильные ответы приведены в конце теста.

1. Укажите наиболее полное описание системы LaTeX.
 - a. LaTeX – компьютерная издательская система, предназначенная для набора математических формул.
 - b. LaTeX - набор макрорасширений системы компьютерной вёрстки TeX, упрощающий набор сложных документов.
 - c. LaTeX – компьютерная издательская система, являющаяся развитием системы TeX.
 - d. LaTeX – набор макрорасширений системы компьютерной вёрстки TeX, предназначенный для верстки научных статей.
2. Какой из перечисленных вариантов не относится к LaTeX?
 - a. Позволяет генерировать содержание и библиографию.
 - b. Имеет специальный синтаксис для оформления математических символов и формул.
 - c. Позволяет вставлять графики, диаграммы, схемы.
 - d. Имеет специальный синтаксис для оформления гипертекстовой разметки.

3. Что понимают под аббревиатурой WYSIWYG?
 - a. Любое приложение с визуальным интерфейсом пользователя.
 - b. Концепция визуального конструирования и отображения документа или приложения на этапе реализации.
 - c. Любое приложение, предназначенное для верстки документов.
 - d. Концепция логической разметки документов.
4. Физическая разметка текста документа предполагает
 - a. его форматирование и форму отображения;
 - b. определение логических ролей каждого фрагмента;
 - c. выделение с помощью команд отдельных фрагментов;
 - d. отделение пустых строк в текстовом редакторе.
5. Что называют дистрибутивом по LaTeX?
 - a. Набор специальных программ для просмотра документов, набранных в LaTeX.
 - b. Текстовые редакторы, предназначенные для верстки документов LaTeX.
 - c. Программы компиляторы, преобразующие текстовую разметку LaTeX в итоговый документ.
 - d. Набор пакетов и программ для работы с файлами формата TEX.
6. Структура документа LaTeX включает следующие основные части:
 - a. преамбулу, описание метаданных и тело документа;
 - b. преамбулу и тело документа;
 - c. заголовок, описание метаданных и тело документа;
 - d. заголовок и тело документа.
7. Для подключения пакетов к документу используется команда
 - a. `\usepackage`;
 - b. `\includepackage`;
 - c. `\insertpackage`;
 - d. `\setpackage`.
8. Какой из перечисленных классов не описан в стандартном наборе классов LaTeX?
 - a. `book`;
 - b. `lecture`;

- c. article;
 - d. letter.
9. Что указывает следующая команда?
- `\usepackage[english,russian]{babel}`
- a. Подключает использование указанных языков.
 - b. Подключает словари грамматической проверки для указанных языков.
 - c. Подключает поддержку указанных языков и режим автопереноса слов.
 - d. Подключает расширенные словари для указанных языков.
10. Укажите символы, относящиеся к символам особого назначения LaTeX:
- a. { } \$ & # % _ ^ ~ \
 - b. { } - № ! % _ : ~ \
 - c. { } () \$ # % _ ^ \
 - d. { } [] & ! % _ : \
11. Для чего предназначена команда `\pagestyle{стиль}`?
- a. Задаёт стиль оформления документа.
 - b. Задаёт стиль оформления страниц.
 - c. Задаёт стиль оформления заголовков.
 - d. Задаёт стиль оформления колонтитулов.
12. Какой из перечисленных форматов имеет текстовый файл с LaTeX-разметкой?
- a. .txt
 - b. .tex
 - c. .latex
 - d. .ltex
13. Для чего может использоваться пакет `geometry`?
- a. Для отображения графических примитивов.
 - b. Для работы построения геометрических фигур.
 - c. Для настройки параметров полей страницы.
 - d. Для разбиения страницы на колонки.
14. Укажите группу команд оформления начертания шрифта, в которой присутствует лишняя:
- a. `\textit`, `\textbf`, `\textsl`, `\emph`;
 - b. `\textit`, `\textbf`, `\textem`, `\textsf`;
 - c. `\textbf`, `\textmd`, `\textup`, `\textsf`;

- d. `\textbf`, `\textsc`, `\textup`, `\textsl`.
- 15. Укажите команду, задающую самый крупный размер шрифта:
 - a. `\Large`;
 - b. `\LargeFontSize`;
 - c. `\Huge`;
 - d. `\HugeFontSize`.
- 16. Для расширенной работы с RGB-моделью требуется подключение пакета
 - a. `colors`;
 - b. `xcolors`;
 - c. `RGBcolors`;
 - d. `RGBxclolors`.
- 17. Какая из команд позволяет начать новый абзац?
 - a. `\`
 - b. `*`
 - c. `\par`
 - d. `\par*`
- 18. В середине текстового абзаца, который выравнивается по ширине, поставлена команда `\`. Какое утверждение является верным в этом случае?
 - a. Текст будет разбит на два абзаца.
 - b. Текст абзаца будет в этом месте разорван, строка до `\` выравнивается по левому краю.
 - c. Текст абзаца будет в этом месте разорван, строка до `\` выравнивается по ширине.
 - d. С абзацем ничего не произойдет.
- 19. Заполнители в LaTeX позволяют
 - a. заполнять пустое пространство строки указанными символами или пробелом;
 - b. заполнять пустое пространство определенным цветом;
 - c. заполнять автоматически пропущенные символы скобок группы `{ }`;
 - d. заполнять автоматически документ специальным текстом-забивкой.
- 20. Отметьте ответ, в котором указан верная иерархия команд рубрикации:
 - a. `\chapter > \part > \section > \subsection`;

- b. `\part > \chapter > \section > \subsection`;
 - c. `\chapter > \paragraph > \subsection > \subsubsection`;
 - d. `\section > \paragraph > \subsection > \subsubsection`.
21. Что будет верным для команды `\chapter*`?
- a. Этот заголовок по умолчанию не вносится в оглавление.
 - b. Это особый заголовок и он является гиперссылкой.
 - c. Для этого заголовка создается сноска.
 - d. Для этого заголовка можно задать особое оформление.
22. Укажите ошибку, допущенную в описании следующего фрагмента кода разметки:
- ```
\usepackage[
 top=2,20cm,
 bottom=2cm,
 left=3cm,
 right=1cm
]{geometry}
```
- a. Команда написана некорректно: нужно написать в одной строке.
  - b. Для значения «2,20» дробную часть нужно отделить точкой.
  - c. У пакета `geometry` не может быть опций.
  - d. Ошибок нет, команда написана верно.
23. Каким образом форматирует изображение следующая команда?
- ```
\includegraphics[width=0.8\textwidth]{work_plan.png}
```
- a. Вставляет изображение шириной 80% от ширины листа.
 - b. Вставляет изображение шириной 80% от ширины текста на странице.
 - c. Вставляет изображение шириной 80% от ширины листа, выравнивает его по центру.
 - d. Вставляет изображение шириной 80% от ширины текста на странице, выравнивает его по центру.
24. Табулятор – это
- a. фиксированное положение каретки, начиная с которого печатается текст;
 - b. команда разбиения фрагмента текста на колонки;

- c. специальная команда, задающая положение текста в ячейке таблицы;
 - d. функция, производящая вычисления по таблице значений.
- 25. Отметьте неверное утверждение для окружения `tabular`.
 - a. Содержимое колонок таблицы нельзя выравнивать по горизонтали.
 - b. Таблицы можно вкладывать.
 - c. Таблицы автоматически переносятся частями на другую страницу.
 - d. Колонки растягиваются пропорционально их содержанию.
- 26. Команда `\cite` позволяет
 - a. вставить перекрестную ссылку на библиографический источник;
 - b. вставить цитату;
 - c. вставить ссылку на веб-ресурс;
 - d. сгенерировать библиографический список.
- 27. Что будет верным для следующей формулы?

$$F(x) = (\sin x + \sin 5x)^2$$
 - a. Печатается отдельной строкой, с выравниванием по центру, автоматически нумеруется.
 - b. Печатается внутри строки абзаца обычным шрифтом.
 - c. Печатается отдельной строкой, с выравниванием по центру.
 - d. Печатается внутри строки абзаца курсивным шрифтом.
- 28. Что понимают под макросом в LaTeX?
 - a. Это специальный пакет, хранящий специальные команды для настройки действия стандартных команд LaTeX.
 - b. Это специальная команда, способная автоматически выполнять определенную последовательность команд.
 - c. Это некоторое символьное имя, закрепленное за последовательностью программных инструкций.
 - d. Это программа, преобразующая код разметки в итоговый документ.
- 29. Какой облачный сервис не позволяет разрабатывать документы в формате LaTeX?
 - a. ShareLaTeX.com.

- b. Overleaf.com.
- c. TeX.StackExchange.com.
- d. LaTeX Base.com.

30. Отметьте верное описание к порядку появления элементов для следующего фрагмента разметки презентации:

```
\begin{frame}[t]{Алгоритм}
  Пункт 1
  \pause
  Пункт 2
  \pause
  Пункт 3
\end{frame}
```

- a. Сначала ни один из пунктов не отобразится, далее появится «Пункт 1», затем «Пункт 2», следом «Пункт 3».
- b. Сначала появится «Пункт 1», через 2 с. – «Пункт 2», через 2 с. – «Пункт 3».
- c. Сначала отобразится «Пункт 1», далее «Пункт 2», следом «Пункт 3».
- d. Сначала ни один из пунктов не отобразится, через 2с. появится «Пункт 1», через 2 с. – «Пункт 2», через 2 с. – «Пункт 3».

Ответы

- | | | | | |
|-------|--------|--------|--------|--------|
| 1. b. | 7. a. | 13. с. | 19. a. | 25. с. |
| 2. d. | 8. b. | 14. b. | 20. b. | 26. a. |
| 3. b. | 9. с. | 15. с. | 21. a. | 27. с. |
| 4. a. | 10. a. | 16. b. | 22. b. | 28. с. |
| 5. d. | 11. d. | 17. с. | 23. b. | 29. с. |
| 6. b. | 12. b. | 18. с. | 24. a. | 30. с. |

3.10 Рекомендации для самостоятельного изучения LaTeX

Читатель уже убедился в многофункциональности LaTeX. Качество и функционал ваших документов можно повысить, подключая новые пакеты. Но пакеты содержат и новые команды, правила их совместного использования, а значит от пользователя потребуется изучение документации, зачастую написанной на английском языке.

3.10.1 Печатные и электронные издания

Обратим внимание на следующие источники. Классикой на русском языке является «Набор и верстка в системе LaTeX» С. М. Львовского [3], [17]; сейчас работы можно найти как в печатной форме, так и в свободном доступе. Если вы успешно прошли наш курс, то смело можете ознакомиться с ней. В качестве недостатка отметим отсутствие в книгах описания работы с изображениями.

Еще одной яркой работой является «LaTeX2ε по-русски», авторы: И. А. Котельников, П. З. Чеботаев [5]. В работе подробно рассматриваются ключевые аспекты системы LaTeX: от работы с текстом до изображений. Учебник хорошо структурирован, качественно оформлен, содержит многочисленные примеры. Распространяется свободно (доступна печатная версия). Можно использовать в качестве учебника и справочника команд одновременно. Однако определенные аспекты требуют от читателя профильного владения ИТ.

К классической литературе по LaTeX относятся работы «LaTeX2ε. Краткое руководство» [29], «Не очень краткое введение в LaTeX2ε» [30], авторы Х. Партль, Э. Шлегл, И. Хина. Оба пособия следует рассматривать как дополнительные источники, и в основном описание касается оформления текста, таблиц и формул. Работы могут быть полезны для изучения определенных особенностей работы с разными языками в LaTeX. Распространяются свободно.

Для более подробного знакомства с элементами LaTeX вам поможет «Компьютерная типография LaTeX» Е. М. Балдина [7]. Здесь можно найти и описание подробностей установки ПО. Теория сопровождается практическими примерами, рассматриваемыми в контексте определенного типа документа. Распространяется свободно (доступна печатная версия).

Хорошей альтернативой источнику [5] служит «TeX для всех» авторов Н. С. Беякова, В. Е. Палоша, П. А. Садовского [8]. Изначально упор делается на оформление текста и рубрикации, материал доступно изложен для широкого класса пользователей. Распространяется свободно (доступна печатная версия).

Полезными могут оказаться и периодически издаваемые методические пособия для студентов и научных работников. В них можно

найти как базовые вопросы работы с LaTeX, так и работу с дополнительными пакетами и классами, например, [18], [19], [20], [23], [24], [33], [36].

Если вы владеете английским языком (достаточно чтения и технического перевода), то используйте иностранные работы из открытых источников. За рубежом LaTeX более популярен и поддерживается обширным коллективом пользователей.

Для глубинного понимания основ работы LaTeX придется познакомиться непосредственно с TeX. Крайне маловероятно, что вам это будет необходимо.

3.10.2 Электронные ресурсы

В сети Интернет читателю не составит труда найти требуемую информацию по LaTeX, в частности справочного характера.

С одной стороны, вы уже познакомились с веб-сервисами **Overleaf.com** и **ShareLaTeX.com**, которые можно было использовать как средство разработки и базу шаблонов разных классов документов. Разумеется, есть и другие сервисы подобного рода, например **LaTeX Base.com**, **PapeerA.com**.

С другой стороны, есть сервисы для решения вспомогательных задач. Отметим **detexify.kirelabs.org**. Он интересен тем, что здесь можно изобразить символ и получить с определенной степенью вероятности его код в LaTeX.

Если вам нужно набрать формулу в хорошем качестве и сохранить ее изображением, например, для другого офисного пакета или сайта (интернет-форума), то используйте следующие сервисы: **www.codecogs.com**, **www.sciweavers.org**.

Многие вопросы, касаемые LaTeX, активно обсуждались и обсуждаются на форумах. Интересным в этом плане является сервис **tex.stackexchange.com**. Пользователи справедливо отмечают: если у вас возник вопрос по LaTeX, значит его уже кто-то задавал на **tex.stackexchange.com**. Ресурс посещает множество пользователей TeX / LaTeX, профессионально владеющих этими системами. Если вы четко опишите вашу проблему и попытку ее решения, то почти наверняка ваш вопрос будет успешно решен. Единственный минус – ресурс англоязычный. Как правило, возникшую проблему можно

найти по ключевым словам в поисковых системах: обычно первые же результаты выдачи ссылаются на tex.stackexchange.com.

3.10.3 Совместимость LaTeX с другими форматами документов

Этот вопрос крайне интерес (вероятно, у читателя он тоже возникал). Можно ли, например, сконвертировать LaTeX документ в MS Word в таком же высоком качестве, или наоборот – экспортировать текст из MS Word для редактирования в LaTeX?

Отвечая на этот и подобные вопросы уже можно догадаться, что прямых инструментов нет. Очевидно, почему: разные концепции набора и оформления, временные рамки реализации этих проектов и конкуренция за пользователя (компания Microsoft заинтересована в развитии коммерческого будущего своего проекта, в то время, как LaTeX – проект с открытым исходным кодом).

Однако далеко не все так плохо! Инструменты конвертирования существуют, пусть и с определенными ограничениями. Остановимся лишь на проекте **IguanaTeX**. Это простой в использовании (и бесплатный) плагин для MS PowerPoint, который позволяет набирать LaTeX-формулы непосредственно в редакторе презентации и отображать его как изображение (на самом деле набирать можно даже часть документа). Можно настроить его качество, скопировать в другое приложение пакета MS Office, а также повторно открыть для редактирования. Внимание! Плагину для работы требуется какой-либо предустановленный дистрибутив по LaTeX (например, MiKTeX).

Мы небезосновательно считаем, что даже при наличии инструментов синхронизации и конвертирования из LaTeX в другие форматы смешивать несколько систем малорационально. Старайтесь выбирать инструмент согласно поставленной задаче и условиям реализации.

ЗАКЛЮЧЕНИЕ

В современной системе образования работа с информационными системами и технологиями становится неотъемлемой частью учебного процесса. Новые условия и социальный запрос информационного общества требует от учителей школ и преподавателей вузов не только владения прикладным программным обеспечением, но и профессионального подхода к его использованию.

Настольная издательская система LaTeX давно зарекомендовала себя в качестве мощного инструмента для разработки документов любого типа. Однако в силу более высокого уровня требований к подготовке специалиста система не находит широкого применения в сфере образования.

В текущем учебном курсе авторы попытались избежать многих недостатков существующих учебных источников и курсов по LaTeX. Мы затронули как базовые вопросы, так и некоторые детали профессиональной подготовки документов, изложили материал простым языком, не вдаваясь в тонкости этой системы и полиграфических стандартов, так, чтобы курс был одинаково доступен специалистам как естественно-научного цикла, так и гуманитарных наук. Главная цель курса была показать сильные стороны автоматизированного оформления, чем и славится LaTeX.

Мы надеемся, что читателю курс оказался интересен и полезен. Возможно вы не будете сталкиваться с LaTeX в дальнейшем. Однако, несомненно, идея системного подхода в построении и оформлении документа поможет вам рациональнее работать с документами в других системах, например, MS Word. Заметим, что последний поддерживает такие средства автоматизации оформления как стили, многоуровневые списки, перекрестные ссылки и многое другое.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Валгина, Н. С.* Теория текста. Учебное пособие / Н. С. Валгина. – М. : Логос, 2003. – 173 с.
2. *Ширинкина, Л. В.* Методы восприятия текста / Л. В. Ширинкина // Актуальные проблемы философии, социологии и политологии, экономики и психологии : 22 Материалы юбилейной научной студенческо-аспирантской конференции (апрель 2001) / Пермский университет. – Пермь, 2001. – Вып. 4 – С. 138-141.
3. *Львовский, С. М.* Набор и верстка в системе LaTeX / С. М. Львовский. – М. : МЦНМО, 2014. – 398 с.
4. *Donald E. Knuth.* The TeXbook. Reading, Massachusetts: Addison-Wesley Publishing Company, 1984. – 494 pp.
5. *Котельников, И. А.* LaTeX2ε по-русски / И. А. Котельников, П. З. Чеботаев. – Новосибирск : Сибирский Хронограф, 2004. – 496 с.
6. Справочная информация по языкам разметки [Электронный ресурс] / Режим доступа:
https://ru.wikipedia.org/wiki/Язык_разметки. – Дата обращения: 12.11.18.
7. *Балдин, Е. М.* Компьютерная типография LaTeX / М. Е. Балдин. – Новосибирск, 2013. – 308 с.
8. *Беляков, Н. С.* TeX для всех / Н. С. Беляков, В. Е. Палош, П. А. Садовский. – М. : ЛИБРОКОМ, 2009. – 208 с.
9. *Кнут, Д. Э.* Все про TeX / Д. Э. Кнут. – М. : Вильямс, 2003. – 560 с.
10. *Кнут, Д. Э.* Компьютерная типография / Д. Э. Кнут. – М. : Мир, 2003. – 686 с.
11. *Donald E. Knuth.* The METAFONTbook. Reading, Massachusetts: Addison-Wesley Publishing Company, 1984. – 375 pp.

12. *Leslie Lamport*. LaTeX: A Document Preparation System. 2nd ed. Reading, Massachusetts: Addison-Wesley Publishing Company, 1994. – 291 pp.
13. *Leslie Lamport*. LaTeX. A Document Preparation System. User's Guide and Reference Manual. Addison-Wesley Publishing Company, 1985. – 342 pp.
14. *Якубович, Д. А.* Основы WEB-разработки-учебно-методическое пособие для проведения лабораторных занятий / Д. А. Якубович, Е. С. Еропова, И. А. Еропов / Владимирский гос. ун-т имени А. Г. и Н. Г. Столетовых. – Владимир : Издательство «Шерлок-пресс», 2017 – 102 с.
15. *Якубович, Д. А.* Пакет прикладных программ MS Word, MS Excel и PowerPoint : учебно-методическое пособие для проведения лабораторных занятий / Д. А. Якубович, Е. С. Еропова, И. А. Еропов / Владимирский гос. ун-т имени А. Г. и Н. Г. Столетовых. – Владимир : Издательство «Шерлок-пресс», 2017. – 60 с.
16. *Якубович, Д. А.* Разработка учебных презентаций средствами MS PowerPoint : учеб.-метод. пособие / Д. А. Якубович, Е. С. Еропова / Мин-во науки и высшего образования Рос. Федерации, ФГБОУ ВО «Владимирский государственный университет им. А. Г. и Н. Г. Столетовых». – Владимир : ВлГУ ; Издательство «Шерлок-пресс», 2019. – 64 с.
17. *Львовский, С. М.* Работа в системе LaTeX / С. М. Львовский. – М. : Национальный Открытый Университет «ИНТУИТ», 2007. – 465 с.
18. *Роженко, А. И.* Искусство верстки в LaTeX'e / А. И. Роженко ; под ред. А. С. Алексеева. – Новосибирск : Издательство ИВМиМГ СО РАН, 2005. – 398 с.
19. *Морозов, Д. К.* Подготовка документов в издательской системе Латех / Д. К. Морозов, А. Я. Пархоменко. – Ярославль : ЯрГУ им. П. Г. Демидова, 2011. – 96 с.
20. *Спивак, М.* Восхитительный TEX: руководство по комфортному изготовлению научных публикаций в пакете AMS-TEX / М. Спивак. – М. : Мир, 1993 – 285 с.
21. An Efficiency Comparison of Document Preparation Systems Used in Academic Research and Development [Электронный ре-

- сурс] / Режим доступа:
<http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0115069/>. – Дата обращения: 7.12.2018.
22. *Brischoux F., Legagneux P.* Don't Format Manuscripts. The Scientist, 23: 24-24. [Электронный ресурс] / Режим доступа: http://www.cebc.cnrs.fr/publipdf/2009/BTS24_2009.pdf. – Дата обращения: 8.12.2018.
 23. *Воронцов, К. В.* LaTeX2ε в примерах / К. В. Воронцов. – М. : Вычислительный центр им. А. А. Дороницина РАН Федерального исследовательского центра «Информатика и управление» Российской академии наук, 2005. – 59 с.
 24. *Грэтцер, Г.* Первые шаги в LaTeX / Г. Грэтцер – М. : Мир, 2000. – 172 с.
 25. *Гуссенс, М.* Путеводитель по пакету LaTeX и его графическим расширениям: иллюстрирование документов при помощи TEX'a и Postscript'a / М. Гуссенс, С. Ратц, Ф. Миттельбах. – М. : Мир, 2002. – 621 с.
 26. *Балдин, Е.* Каталог классов и стилей LaTeX. Часть 4. Подготовка научных публикаций [Электронный ресурс] / Е. Балдин // Режим доступа:
https://www.ibm.com/developerworks/ru/library/latex_styles_04/. – Дата обращения: 6.02.2019.
 27. Каталог CTAN [Электронный ресурс] / Режим доступа: <http://texcatalogue.ctan.org/>. – Дата обращения: 10.02.2019.
 28. Облачный сервис для разработки документов на базе LaTeX Overleaf.com [Электронный ресурс] / Режим доступа: <https://www.overleaf.com/>. – Дата обращения: 5.09.2018.
 29. *Партль, Х.* LaTeX2ε. Краткое руководство / Х. Партль, Э. Шлегл, И. Хина ; пер. с нем. Р. В. Загретдинова ; под ред. И. А. Маховой, 1998. – 43 с.
 30. *Партль, Х.* Не очень краткое введение в LaTeX2ε. / Х. Партль, Э. Шлегл, И. Хина; пер. с нем. Б. Тоботраса. – М., 2003. – 153 с.
 31. *Давлетярова, Е. П.* Издательская система LaTeX. Методические указания к выполнению практикума по курсу «Информационные технологии в математике» / Е. П. Давлетярова,

- А. В. Шутов, Ю. А. Медведев. – Владимир : ВГГУ, 2009. – 44 с.
32. *Столяров, А. В.* Сверстай диплом красиво : LATEX за три дня / А. В. Столяров – М. : МАКС Пресс, 2010. – 100 с.
33. *Чебарыков, М. С.* Основы работы в системе LATEX : учебное пособие для студентов направления «Информатика и вычислительная техника» / М. С. Чебарыков ; Рубцовский индустриальный институт. – Рубцовск, 2014. – 49 с.
34. *Драгунов, Т. Н.* Презентации в LaTeX: учебное пособие / Т. Н. Драгунов, С. А. Королев, А. Д. Морозов. – М.-Ижевск : НИЦ «Регулярная и хаотичная динамика», 2009. – 94 с.
35. *Жидков, А. А.* Интерактивные презентации в системе LaTeX: учебно-методическое пособие / А. А. Жидков. – Нижний Новгород : Издательство Нижегородского госуниверситета, 2010. – 41 с.
36. *Ширяева, Е. В.* Введение в TeX-программирование / Е. В. Ширяева, И. В. Ширяева. – Ростов-на-Дону, 2010. – 115 с.
37. *Braams, J.* Babel, a multilingual package for use with LaTeX's standart document classes [Электронный ресурс] / J. Braams, 2004. – 452 pp. – Режим доступа:
<http://www.pvv.ntnu.no/~berland/latex/docs/babel.pdf>. – Дата обращения: 13.03.2019.
38. *Rahtz, S.* Hypertext marks in LaTeX: a manual for hyperref [Электронный ресурс]/ S. Rahtz, Н. Oberdiek, 2008. – 27 pp. – Режим доступа:
<http://www.pd.infn.it/TeX/doc/latex/hyperref/manual.pdf>. – Дата обращения: 21.04.2019.
39. Справочная информация по языкам разметки [Электронный ресурс] / Режим доступа:
https://ru.wikipedia.org/wiki/Язык_разметки. – Дата обращения: 12.11.18.
40. Справочная информация по MS Word [Электронный ресурс] / Режим доступа:
https://ru.wikipedia.org/wiki/Microsoft_Word. – Дата обращения: 7.10.18.

41. Справочная информация по MS PowerPoint [Электронный ресурс] / Режим доступа:
https://ru.wikipedia.org/wiki/Microsoft_PowerPoint. – Дата обращения: 7.10.18.
42. Справочная информация по MS Publisher [Электронный ресурс] / Режим доступа:
https://ru.wikipedia.org/wiki/Microsoft_Publisher. – Дата обращения: 7.10.18.
43. Описание продуктов Apache OpenOffice [Электронный ресурс] / Режим доступа:
<https://www.openoffice.org/ru/>. – Дата обращения: 8.10.18.
44. Описание системы Adobe InDesign [Электронный ресурс] / Режим доступа:
<https://www.adobe.com/ru/products/indesign.html>. – Дата обращения: 13.10.18.
45. Информация о наборе формул на портале Wikipedia [Электронный ресурс] / Режим доступа:
<https://ru.wikipedia.org/wiki/Википедия:Формулы>. – Дата обращения: 28.04.19.
46. Правила русской орфографии и пунктуации : полный академический справочник / под ред. В. В. Лопатина ; Российская академия наук, Отделение историко-филологических наук, Институт русского языка им. В. В. Виноградова. – М. : Эксмо, 2007. – 478 с.
47. Приказ об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки) (уровень бакалавриата). Регистрационный №41305 от 02.03.2016 [Электронный ресурс] / Режим доступа: <https://www.fgosvo.ru/uploadfiles/fgosvob/440305.pdf>. – Дата обращения: 8.01.18.
48. Дистрибутив TeX Live [Электронный ресурс] / Режим доступа:
<https://www.tug.org/texlive/>. – Дата обращения: 8.12.18.
49. Дистрибутив MiKTeX [Электронный ресурс] / Режим доступа:
<https://miktex.org/>. – Дата обращения: 8.12.18.

50. Дистрибутив MacTeX [Электронный ресурс] / Режим доступа:
<http://www.tug.org/mactex/>. – Дата обращения: 8.12.18.
51. Редактор TeXstudio [Электронный ресурс] / Режим доступа:
<http://www.texstudio.org/>. – Дата обращения: 9.12.18.
52. Редактор WinEdt [Электронный ресурс] / Режим доступа:
<http://www.winedt.com/>. – Дата обращения: 9.12.18.

ГЛОССАРИЙ

А

Абзац (текстовый абзац) – фрагмент письменной или печатной речи, состоящий из одного или нескольких предложений. По правилам полиграфии абзацы имеют завершённый логический смысл и, как правило, визуально отделяются друг от друга вертикальными интервалами, с отступом красной строки.

Аккаунт (учетная запись) – запись в базе данных ресурса или приложения, хранящая сведения о пользователе и его правах.

Б

Библиографическая база данных – электронная система, предназначенная для размещения, хранения, поиска и обмена данными, а также позволяющая использовать ссылки на ее источники. В рамках документа библиографическая база – это оформленный по определенным ключам список источников.

В

Веб-ресурс (интернет-ресурс) – это информационная система, использующая веб-технологии и предназначенная для оказания публичных информационных услуг в сети Интернет.

Визуальный редактор (WYSIWYG-редактор) – это программное средство для реализации работы по визуальному оформлению страниц документа, шаблонов и материалов по средствам технологии WYSIWYG (от англ. What You See Is What You Get, «что видишь, то и получишь»).

Верстка (компьютерная верстка) – процесс разработки документа. В полиграфии под версткой также понимают процесс формирования полосы издания и оттиск, направляемый на корректуру.

Г

Гарнитура (типографика) – набор из одного или нескольких шрифтов в одном или нескольких размерах и начертаниях, имеющих стилевое единство рисунка и состоящих из определённого набора типографских знаков.

Гипертекст – это система текстовых страниц, имеющих перекрёстные ссылки. На базе гипертекстовой технологии организованы современные веб-ресурсы.

Группа (в LaTeX) – это часть текста разметки, заключенная в фигурные скобки и имеющая определенное внутри нее форматирование.

Д

Декларация (в LaTeX) – это команда, меняющая оформление текста или иных объектов с момента ее указания. Область действия декларации можно ограничивать скобками группы, либо отменять другими декларациями.

Дистрибутив – это форма распространения программного обеспечения. В отличие от исходного программного обеспечения, дистрибутив включает специальные файлы и интерфейс для его начальной установки и настройки.

З

Заполнитель (в LaTeX) – это специальная команда, позволяющая расталкивать два или более объекта по левой / правой или верхней / нижней границам листа (блока). Для горизонтальных заполнителей возможно указание заполняющих символов (линии, точки).

И

Интегрированная среда разработки (ПО, документов) – это комплекс программных средств прикладного назначения, используемый верстальщиками и программистами для разработки документов и программного обеспечения соответственно.

Интерлиньяж – междустрочный пробел, расстояние между базовыми линиями соседних строк. В общепринятом смысле под этим термином понимают «межстрочный интервал».

К

Кегль – это размера шрифта.

Класс документа (в LaTeX) – это набор стилевых настроек, определяющих шаблон оформления документа.

Кодировка (текста) – специальный стандарт, определяющий способ кодирования текстовых символов и объем памяти, отводимый на хранение одного символа. Чем больше последний, тем больше символов можно закодировать.

Компилятор (программа компилятор) – программа, выполняющая трансляцию программы, составленной на языке высокого уровня, в эквивалентную программу на низкоуровневом языке, близком машинному коду.

Компоновка документа – это расположение и структуризация отдельных частей документа. Может осуществлять автоматически, по указанному макету или разметке.

Конструктор сайтов – прикладное программное обеспечение, реализующее инструменты по разработке веб-страниц в WYSIWYG-конструкторе.

Кракозябры – символы, отображающиеся неверно. Основная причина их появления – выбор неверной кодировки при сохранении файла или при его просмотре в редакторе.

Л

Логическая разметка (текста, документа) – это разметка, определяющая роль каждого фрагмента текста в общей структуре документа (например, заголовок, текстовый абзац, таблица, изображение, другое).

М

Макет документа – шаблон стилевых настроек, задающих параметры отображения документа в целом и его отдельных элементов на страницах. В визуальных редакторах макет настраивается с помощью специальных инструментов визуального редактирования, в языках разметки – с помощью файлов, хранящих метаданные оформления.

Макрос (в LaTeX) – это некоторое символьное имя, закреплённое за последовательностью (как правило, логически связанных) программных инструкций. Макрос не является командой LaTeX, он лишь скрывает часть команд на этапе разработки документа в более короткой и удобной для использования форме.

Марашка – дефект печати в виде мелких, визуально-заметных на оттиске следов краски, печатных элементов, или отсутствие элементов изображения. Обычно марашки возникают при попадании на печатную форму или на офсетную резинотканевую пластину посторонних частиц (кусочков бумаги, пыли, засохшей краски и пр.).

Метаданные – информация о другой информации или данных. Метаданные содержат описание сведений, характеристик, структуры и других свойств сущности, что позволяет автоматизировать обработку этих данных.

Н

Неразрывный пробел – это пробел между двумя текстовыми элементами, который не позволяет разорвать их на несколько строк. Неразрывный пробел обычно ставится для предотвращения разрыва ФИО, названий, ссылок на объект в документе и связанных с ними номеров.

Настольная издательская система – комплект оборудования для подготовки оригинал-макета издания, готового для передачи в типографию.

Начертание (шрифта) определяет стиль оформления букв шрифта.

О

Облачные технологии (сервисы, ресурсы, вычисления) – это модель обеспечения удобного сетевого доступа к некоторому фонду конфигурируемых вычислительных ресурсов, которые могут быть оперативно предоставлены и освобождены с минимальными затратами или обращениями к провайдеру. Для работы с облачным сервисом требуется подключение к сети Интернет. Услуга доступа может предоставляться бесплатно, при регистрации, платно или по иным условиям.

Окружение (в LaTeX) – это команда, осуществляющая настройку и форматирование элементов, указанных внутри ее тела.

Опции (в LaTeX) – необязательные параметры команды или окружения, обозначаемые в квадратных скобках и указывающие, как осуществить действие над параметром.

П

Пакет (в LaTeX) – специальный файл, хранящий описание дополнительных команд и макросов. Пакеты подключаются в к основному файлу и предоставляют упрощенные возможности для реализации определенных задач оформления документа.

Параметр (в LaTeX) – специальная команда, задающая значение определенному атрибуту документа. Параметры можно рассматривать в качестве стандартных переменных величин в LaTeX.

Перекрестная ссылка – это специальный элемент, хранящий метаданные элемента и позволяющий автоматически ссылаться на указанный меткой элемент документа.

Подписной лист – оттиск, отпечатанный по окончании всех подготовительных технологических операций и полностью соответствующий требованиям заказа. Утверждается мастером печатного цеха, представителем заказчика или другим ответственным лицом и является эталоном при печати тиража издания.

Полиграфия – отрасль промышленности, занимающаяся изготовлением печатной продукции: книжно-журнальной, деловой, газетной, этикеточной и упаковочной продукции.

Преамбула документа (в LaTeX) – это часть разметки документа, размещающая информацию о типе документа, его структуре, оформлении, данные для ключевых разделов, подключаемые классы и иные метаданные.

Программное обеспечение (ПО) – это программа или совокупность (комплекс) программ, используемых для управления работы компьютером.

Р

Разрыв строки – специальная команда, осуществляющая перенос текста абзаца на новую строку. Разорванная строка выравнивается согласно заданному правилу (по ширине, по левому краю). Разрыв строки не начинает новый абзац.

Рубрикация – это разбиение текста на главы, разделы, параграфы и пункты. Рубрикация необходима для автоматизации генерации оглавления, навигации, а также для предоставления метаданных системам сбора информации (например, поисковым системам).

С

Средство просмотра – программа для просмотра итогового вида документа, набранного с помощью языка разметки, или являющаяся частью интерфейса визуального реактора документа.

Т

Табулятор – это фиксированное положение каретки, начиная с которого печатается текст. В строку можно установить несколько табуляторов. Табулирование позволяет организовать простейшее выравнивание текстовых данных в форме таблицы.

Тег (в языках разметки) – специальная команда, определяющая логический смысл элемента или его оформление.

Тело документа (в LaTeX) – часть разметки документа, содержащая текст и команды, определяющие его логическую структуру и форматирование.

Текстовый процессор – компьютерная программа, используемая для набора и модификации документов, компоновки макета текста и предварительного просмотра документа в режиме визуального редактора.

Текстовый редактор – компьютерная программа или часть программного продукта (например, редактор исходного кода интегрированной среды разработки или окно ввода в браузере), предназначенная для создания и изменения текстовых данных и файлов.

Ф

Физическая разметка (текста, документа) – это разметка, определяющая форму отображения (форматирование) элемента документа (например, полужирный, подчеркнутый, красным цветом, с отступами слева и справа).

Ц

Цветовая модель CMYK – модель, которая описывает цвет путем смешения полиграфический красок: C – Cyan (голубой); M – Magenta (пурпурной); Y – Yellow (желтой); а также черной – K – Key color (по одной версии) или black – K (по другой версии).

Цветовая модель RGB (аббревиатура английских слов red (красный), green (зеленый), blue (синий)) – это трехцветная модель, состоящая из основных цветов радужного спектра.

Цифровая печать – технология получения оттисков в печатной машине с использованием переменной печатной формы, изменениями в которой при каждом цикле управляет компьютер издательской системы. Этот вид техники используют для малотиражных рекламных или коммерческих изданий, в которые должны быть внесены измене-

ния в процессе изготовления тиража. В некоторых машинах возможно внесение изменений после печатания даже одного экземпляра.

Ш

Шаблон (оформления документа) – набор стилевых настроек, отвечающих за оформление содержимого документа определенным образом. В языках разметки шаблон задается либо непосредственно скопированной частью разметки, либо подключаемыми пакетами.

Я

Язык разметки (текста) – это набор символов или последовательностей, содержащийся в тексте и передающий информацию о его структуре и форматировании. Выделяют логическую и физическую разметку.

При подготовке глоссария использовались следующие ресурсы:

- Глоссарий полиграфических терминов:
<http://www.colorprint.ru/index.php?id=5>.
- Интернет энциклопедия Wikipedia.org:
<https://ru.wikipedia.org/>.

ОБ АВТОРАХ

Якубович Денис Андреевич – старший преподаватель кафедры математического образования и информационных технологий Владимирского государственного университета.

Сфера научных интересов: дифференциальные уравнения в математической физике.

Преподаваемые дисциплины: «Информационные технологии», «Веб-технологии», «Практикум по решению задач на ЭВМ», «Основы математической обработки информации».

E-mail: yakubovich.studylib@mail.ru

Еропова Елена Станиславовна – кандидат педагогических наук доцент кафедры математического образования и информационных технологий Владимирского государственного университета.

Сфера научных интересов: проблемы информационных технологий в образовании.

Преподаваемые дисциплины: «Информационные технологии», «Информационные технологии в образовании», «Основы математической обработки информации».

E-mail: eropova13061962@mail.ru

Учебное электронное издание

ЯКУБОВИЧ Денис Андреевич
ЕРОПОВА Елена Станиславовна

ИЗДАТЕЛЬСКАЯ СИСТЕМА LaTeX

Учебное пособие

Издается в авторской редакции

Системные требования: Intel от 1,3 ГГц; Windows XP/7/8/10; Adobe Reader; диск-вод CD-ROM.

Тираж 27 экз.

Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых
Изд-во ВлГУ
rio.vlgu@yandex.ru