

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

УТВЕРЖДАЮ

Проректор
по учебно-методической работе

А.А.Панфилов

« 15 » 02 2016г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Системное программное обеспечение

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Профиль/программа подготовки:

Уровень высшего образования: бакалавриат

Форма обучения: Очная

Семестр	Трудоем- кость зач. ед./ час	Лекций, час.	Практич. занятий, час.	Лаборат. работ, час	СРС, час.	Форма промежу- точного контроля (экл./зачет)
4	6/216	36	18	36	90	Экзамен 36
5	5/180	36	18	18	72	Экзамен + КР (36)
Итого	11/396	72	36	54	162	зачет/экзамен + КР(36)

Владимир 2016

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целями освоения дисциплины (модуля) «Системное программное обеспечение» являются:

1. Обучение студентов основам построения компиляторов, общим принципам программирования, представления основных структур программ и данных.
2. Создание фундаментальной основы знаний, необходимой при проектировании программных продуктов для вычислительных систем.
3. Изучение основных этапов и фаз построения компиляторов.
4. Изучение синтаксических и семантических конструкций языков программирования высокого уровня.
5. Освоение студентами основных приемов программирования типовых задач (списки, стеки, очереди, деревья).
6. Изучение особенностей реализации обработки различных типов данных.
7. Освоение студентами соответствия между операторами языков высокого уровня и командами языка ассемблера (фаза генерации кода).

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Системное программное обеспечение» относится к вариативной части ОПОП по направлению 09.03.01 – «Информатика и вычислительная техника» бакалавриат.

Дисциплина логически, содержательно и методически тесно связана с рядом теоретических дисциплин и практик ОПОП.

Для успешного изучения дисциплины «Системное программное обеспечение» студенты должны быть знакомы с дисциплинами базовой части: «Математика», «Информатика», «Программирование», «Дискретная математика и математическая логика», которые формируют необходимые для изучения данной дисциплины способности к обобщению и анализу информации, знания математического анализа и алгоритмов, структурных блоков ЭВМ, способов представления данных в ЭВМ, способность использовать персональный компьютер и системы программирования для разработки программного обеспечения, готовность понимать актуальность совершенствования языков программирования, подходов к проектированию программных систем, программного обеспечения в аспектах технического и научного прогресса.

Без освоения дисциплины «Системное программное обеспечение» невозможна дальнейшая успешная подготовка студентов по направлению 090301. Дисциплина предоставляет саму возможность изучения практически всех последующих дисциплин профессионального цикла, поскольку в процессе изучения используются ЭВМ и языки высокого уровня, как средства и инструменты

для исследований и получения результатов, для решения специализированных задач.

Дисциплина является основой следующих дисциплин базовой и вариативной части: «Технология программирования», «Операционные системы», «Базы данных», «Инженерная и компьютерная графика», «Организация ЭВМ и систем», и ряде других дисциплин, связанных с изучением или использованием программного обеспечения ЭВМ.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

В результате изучения дисциплины студенты должны:

Изучить и исследовать в интерактивном режиме: вопросы решения стандартных задачах профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учётом основных требований информационной безопасности (ОПК-5), разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов "человек - электронно-вычислительная машина" (ПК-1).

Знать: основные концепции системного программирования; структуры данных, принципы построения трансляторов, основные этапы и фазы процесса компиляции.

2) **Уметь:** осваивать методики использования программных средств для решения практических задач; разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования; обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности; настраивать и налаживать программно – аппаратные комплексы; устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем.

3) **Владеть:** способностью к обобщению, анализу, восприятию информации, постановке цели и выбору путей её достижения; способностью находить организационно-управленческие решения в нестандартных ситуациях; основными законами естественнонаучных дисциплин в профессиональной деятельности, применять методы математического анализа и моделирования, теоретического и экспериментального исследования; навыками работы с компьютером как средством управления информацией; способностью работать с информацией в глобальных компьютерных сетях; владеть одним из иностранных языков на уровне не ниже разговорного.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Системное программное обеспечение» составляет 11 зачетных единиц, 396 часов.

№ п/п	Раздел (тема) дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студен- тов и трудоемкость (в часах)							Объем учебной работы, с применением интерактивных методов (в часах / %)	Формы текущего контроля успеваемо- сти (по неделям семест- ра) , форма промежуточ- ной аттестации (по семестрам)
				Лекции	Семинары	Практические занятия	Лабораторные работы	Контрольные работы, колло- квиумы СРС	КП / КР			
1	Основные принци- пы по- строения трансля- торов	4	1- 2	4					10		2 часа/50%	
2	Таблицы иденти- фикато- ров	4	3- 4	4		4			15		4 часа/50%	Рейтинг-контроль №1
3	Лексиче- ский ана- лиз	4	5- 8	8		4			20		6 часов/50%	
4	Синтак- сический анализ	4	9- 12	8		4	8		15		6 часов/35%	Рейтинг-контроль №2
5	Семанти- ческий анализ	4	13 - 14	4		4	8		10		4часа/25%	
6	Распреде- ление па- мяти	4	15 - 16	4			10		10		6 часов/75%	
7	Генера- ция и оп- тимиза- ция кода	4	17 - 18	4		2	10		10		8часа/50%	Рейтинг-контроль №3
Итого за се- местр		4	18	36		18	36		90		36 часов/40%	экзамен
1	Проекти- рование компиля- тора	5	1- 2	4		2			8		2 часа/33%	
2	Внешний проект	5	3	2					8		2 часа/100%	
3	Проект архитек- туры	5	4	2		2			8		8 часов/20%	Рейтинг-контроль №1
4	Трансля- торы с языка ассембле- ра	5	5- 6	4		2			8		4 часа/66%	

5	Средства ассембле- ра	5	7- 8	4		2			8		4 часа/66%	
6	Структу- ра совре- менной системы програм- мирова- ния	5	9- 10	4		2	4		10		4 часа/40%	Рейтинг-контроль №2
7	Принци- пы функ- циониро- вания систем програм- мирова- ния	5	11 - 14	8		4	4		6		4 часа/25%	
8	Абсолют- ный и началь- ный за- грузчики	5	15 - 16	4		2	4		8		4 часа/40%	
9	Настраи- вающий и связы- вающий загрузчи- ки	5	17 - 18	4		2	6		8		4 часа/33,3%	Рейтинг-контроль №3
Итого за се- местр		5	18	36		18	18		72		36 часов/50%	экзамен

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При освоении дисциплины используются следующие сочетания видов учебной работы с методами и формами активизации познавательной деятельности бакалавров для достижения запланированных результатов обучения и формирования компетенций:

- изучение теоретического материала дисциплины на лекциях с использованием компьютерных технологий;
- самостоятельное изучение теоретического материала дисциплины с использованием Интернет-ресурсов, информационных баз, методических разработок, специальной учебной и научной литературы;
- закрепление теоретического материала при проведении лабораторных работ с использованием современной вычислительной техники и систем программирования, выполнения проблемно-ориентированных, поисковых, творческих заданий;
- самостоятельная работа студента, направленная на углубление и закрепление знаний, развитие практических умений, комплекса универсальных (общекультурных) и профессиональных компетенций, повышение творческого потенциала, заключается в работе с лекционным материалом, поиске и анализе литературы и электронных источников информации по заданной

проблеме, переводе материалов из тематических информационных ресурсов с иностранных языков, изучении тем, вынесенных на самостоятельную проработку, подготовке докладов и презентаций по результатам выполненной работы, изучении теоретического материала к лабораторным занятиям, подготовке к экзамену.

Удельный вес занятий, проводимых в интерактивной форме составляет 50% от аудиторной нагрузки.

6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО РЕЙТИНГ КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

4 семестр

6.1. Оценочные средства для текущего контроля успеваемости

а) отчет по выполненным лабораторным работам.

Лабораторный практикум является групповой аудиторной работой в малых группах. Целью лабораторного практикума является:

- подтверждение теоретического материала, полученного на лекционных занятиях, путем разработки небольших по объему программ по изучаемой теме в условиях лабораторий вуза или сторонних предприятий;

- приобретение практических навыков и компетенций в области постановки и решения задач по профилю профессиональной деятельности.

Перед проведением лабораторных занятий студенты должны освоить требуемый теоретический материал и процедуры выполнения лабораторной работы по выданным им предварительно учебным и методическим материалам.

Перечень тем лабораторных работ:

1. Списки. Представление однородными массивами фиксированного размера.
2. Списки. Реализация с помощью динамических переменных.
3. Бинарные деревья. Итеративный алгоритм обработки.
4. Бинарные деревья. Рекурсивный алгоритм обработки.

б) на практических занятиях студенты осваивают основные процедуры компиляции и методы их создания на примере проектирования компилятора, начиная с грамматики для математического выражения, заканчивая процедурами генерации кода на языке Ассемблер (и его выполнение) и непосредственного вычисления значения выражения по дереву операций. Оценивают правильность результатов работы последних процедур. Анализируют сгенерированный код с целью его оптимизации.

Перечень тем практических занятий:

1. Лексема. Таблица лексем. Преобразование программы на ЯВУ в список лексем.
2. Лексический анализ. Сканер на основе детерминированного автомата.
3. Сканер на основе инженерной импровизации.
4. Диаграммы Вирта. БНФ. РБНФ.
5. Грамматика арифметических выражений. Нисходящий и восходящий синтаксический анализ. Дерево разбора.
6. Внутреннее представление программы: обратная польская запись, семантическое дерево (дерево операций), тетрады и триады.
7. Грамматики с левосторонней и правосторонней рекурсией. Процедура синтаксического анализа.
8. Процедура генерации кода. Вычисление значения выражения по дереву операций.

в) устный или письменный опрос студентов во время лекции по изучаемому материалу.

6.2. Оценочные средства для текущего контроля

Рейтинг-контроль №1

1. Транслятор, компилятор, интерпретатор. Особенности и отличия.
2. Общая схема работы компилятора. Фазы компиляции.
3. Компилятор можно построить не используя лексический анализатор (ЛА). Почему большинство современных компиляторов все-таки используют фазу ЛА?
4. Какую роль выполняет ЛА в процессе компиляции? Как могут быть связаны между собой лексический и синтаксический анализы?
5. Что такое таблица идентификаторов (ТИ)? Для чего она предназначена? Какие существуют способы организации ТИ?
6. Какие цели преследуются при организации таблиц идентификаторов (ТИ)? Исходя из каких характеристик оценивается эффективность того или иного метода организации ТИ?
7. Чем различаются таблица лексем и ТИ?
8. Что такое хэш-функция и хеш-адресация? Для чего они используются?
9. Что такое коллизия? Почему она происходит при использовании хэш-функции? В чем заключаются преимущества и недостатки метода цепочек?
10. Что такое «метод дихотомии»? Для чего он используется и в чем заключаются особенности его использования?
11. Как могут быть скомбинированы различные методы организации ТИ?
12. От чего зависит количество проходов, необходимых компилятору для построения результирующей объектной программы? Как влияют на количество проходов синтаксис входного языка программирования, семантика этого языка и архитектура целевой вычислительной системы?

13. Построить семантическое дерево след. выражения:

1. $A+B*(C-D)/F$

2. $A-B/C+D*F$

а) основываясь на грамматике с левосторонней рекурсией

б) основываясь на грамматике с правосторонней рекурсией.

Рейтинг-контроль №2

1. Фаза синтаксического анализа. Задачи синтаксического анализатора (СА).
2. Виды СА.
3. Дайте характеристику восходящего СА.
4. Дайте характеристику нисходящего СА.
5. Для каких грамматик применим синтаксический анализ методом рекурсивного спуска? Опишите этот метод.
6. Для каких грамматик применим синтаксический анализ методом «сдвиг-свертка»? Опишите этот метод.
7. Обработка синтаксических ошибок. Виды ошибок.
8. Стратегии восстановления после ошибок.
9. Способы внутреннего представления входной программы.
10. Фаза семантического анализа. Этапы семантического анализа.

Рейтинг-контроль № 3

1. Принципы распределения памяти.
2. Виды переменных. Классификация областей памяти.
3. Статическое и динамическое связывание. Менеджер памяти.
4. Дисплей памяти процедуры.
5. Стандартные соглашения о связывании подпрограмм
6. Фаза генерации кода. Общие принципы генерации кода.
7. Суть и принципы синтаксически управляемого (СУ)-перевода.
8. Приведите пример генерации кода по дереву операций.
9. Приведите пример преобразование дерева операций в последовательность триад
10. Цели фазы оптимизации кода.
11. Машинно-независимые и машинно-зависимые методы оптимизации.
12. Перечислите возможные операции по оптимизации линейных участков программы.
13. Опишите операцию «свертка объектного кода».
14. Опишите операцию «исключение лишних операций».

6.3. Промежуточная аттестация

Список экзаменационных вопросов

1. Классификация структур данных
2. Линейные списки
3. Связное представление списков
4. Операции со списками.
5. Список свободного пространства
6. Стеки. Очереди. Деки.

7. Многомерные структуры данных переменного размера
8. Деревья. Основные определения.
9. Формы представления деревьев
10. Бинарные деревья. Основные особенности.
11. Алгоритм преобразования любого дерева в бинарное.
12. Процедуры сканирования деревьев.
13. Обратная польская запись.
14. Алгоритм Дейкстры.
15. Статическая, автоматическая, динамическая память.
16. Фазы управления памятью.
17. Применение стека для управления автоматической памятью.
18. Применение связанных списков для управления динамической памятью.
19. Алгоритмы резервирования памяти.
20. Фрагментация памяти. Способы ее устранения.
21. Процесс сборки мусора.
22. Отслеживание свободной памяти с помощью подсчета ссылок.
23. Отслеживание свободной памяти с помощью разметки

6.4. Самостоятельная работа студентов

Перечень тем для СРС

1. Основные принципы построения трансляторов.
2. Структуры данных.
3. Бинарные деревья.
4. Лексический анализ. Таблица лексем.
5. Синтаксический анализ. Дерево грамматического разбора.
6. Семантический анализ
7. Фаза «Подготовка к генерации кода»
8. Генерация и оптимизация кода.

5 семестр

6.5. Оценочные средства для текущего контроля успеваемости

а) Перечень тем лабораторных работ

1. Таблицы идентификаторов. Принципы построения.
2. Обработчик ошибок.
3. Построение лексического анализатора.
4. построения синтаксического анализатора.

б) Перечень тем практических занятий

1. Внешний проект компилятора подмножества языка Паскаль.
2. Проект архитектуры компилятора подмножества языка Паскаль. Структуры данных, используемых компилятором.
3. Тестирование компилятора. Проектирование тестов.
4. Алгоритм работы и структуры данных ассемблера.
5. Макроопределение и макровывод. Позиционные и ключевые параметры.
6. Условная макрогенерация. Препроцессор языка С.

7. Алгоритм работы и структуры данных макропроцессора.
8. Алгоритмы работы абсолютного и настраивающего загрузчиков.
9. Компоновщик и связывающий загрузчик. Алгоритм работы связывающего загрузчика

в) устный или письменный опрос студентов во время лекции по изучаемому материалу.

6.6. Оценочные средства для текущего контроля

Рейтинг-контроль № 1

1. Ассемблеры. Назначение. Алгоритмы работы.
2. Макропроцессор. Определение и назначение.
3. Структуры данных и логика работы макропроцессора.
4. Макропроцессор: ключевые параметры.
5. Макропроцессор: позиционные параметры.
6. Условная макрогенерация
7. Функции препроцессора. Препроцессор языка С
8. Структура системы программирования.
9. Принципы функционирования системы программирования.
10. Управляющие секции и связывание модулей.
11. Библиотеки подпрограмм.

Рейтинг-контроль № 2

1. Абсолютный загрузчик. Функции и алгоритм работы.
2. Виды загрузчиков. Начальный загрузчик.
3. Что такое трансляция адресов? Как она выполняется?
4. Перемещающий загрузчик. Функции и алгоритм работы.
5. Связывающий загрузчик. Функции и логика работы.
6. Что такое компоновщик? Чем он отличается от связывающего загрузчика?
7. Управление программой с оверлейной структурой.
Управление программой с динамической структурой. Динамический загрузчик. Функции и логика работы.

Рейтинг-контроль № 3

1. Схема вывода.
2. Понятие грамматического разбора. Дерево разбора.
3. Нисходящий и восходящий грамматический разбор.
4. Общая схема работы компилятора.
5. Транслятор, компилятор, интерпретатор – отличия и особенности.
6. Фазы компиляции.
7. Фаза лексического анализа.
8. Таблица идентификаторов.
9. Способы организации таблиц идентификаторов.
10. Хэш-функция. Хэш-адресация.

11. Методы сортировки.
Фаза синтаксического анализа.

6.7. Промежуточная аттестация

Список экзаменационных вопросов

1. Формальные языки. Основные определения.
2. Формальные грамматики. Основные определения.
3. Классификация грамматик Хомского.
4. КС-грамматики и КЗ-грамматики.
5. Регулярные грамматики.
6. Грамматика с левосторонней и с правосторонней рекурсией.
7. Схема вывода.
8. Понятие грамматического разбора. Дерево разбора.
9. Нисходящий и восходящий грамматический разбор.
10. Общая схема работы компилятора.
11. Транслятор, компилятор, интерпретатор – отличия и особенности.
12. Фазы компиляции.
13. Фаза лексического анализа.
14. Таблица идентификаторов.
15. Способы организации таблиц идентификаторов.
16. Фаза синтаксического анализа.
17. Обработка синтаксических ошибок. Виды ошибок.
18. Стратегии восстановления после ошибок.
19. Способы внутреннего представления входной программы.
20. Фаза семантического анализа.
21. Фаза генерации кода.
22. Принципы распределения памяти.
23. Виды переменных. Классификация областей памяти.
24. Статическое и динамическое связывание. Менеджер памяти.
25. Дисплей памяти процедуры.
26. Стандартные соглашения о связывании подпрограмм.
27. Фаза генерации кода. Общие принципы генерации кода.
28. Суть и принципы синтаксически управляемого (СУ)-перевода
Фаза оптимизации кода.

6.8. Курсовая работа

Типовое задание на курсовую работу предполагает разработку системы компиляции. Входным языком является подмножество языка Pascal определяемое индивидуальным вариантом.

В ходе выполнения курсовой работы должен сформироваться проект. Проект (projectus - брошенный вперёд) - система документов, определяющая процесс создания какого-либо продукта (его бумажная модель). Здесь за основу принят структурный подход с его методом пошаговой детализации, что вполне целесо-

образно в применении к программным системам (ПС) малого и даже среднего размера (до 10 т. операторов).

**Варианты индивидуальных заданий для курсового проектирования
(характеризует набор конструкций входного языка):**

1. Массив, FOR, IF, BREAK
2. Массив, WHILE, сравнение.
3. Массив, REPEAT, сравнение.
4. Метка, IF, сравнение, GOTO.
5. Константа, CASE, GOTO, метка.
6. Процедура, EXIT, передача данных по указателю, константа.
7. Функция, передача данных по указателю, константа.
8. IF, сравнение, OR, AND.
9. IF, константа, XOR, NAND.

6.9. Самостоятельная работа студентов

Перечень тем для СРС

1. Проектирование компилятора
2. Внешний проект
3. Проект архитектуры
4. Трансляторы с языка ассемблера
5. Средства ассемблера
6. Структура современной системы программирования
7. Принципы функционирования систем программирования
8. Абсолютный и начальный загрузчики
9. Настраивающий и связывающий загрузчики

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ «Системное программное обеспечение»

7.1. Основная литература

1. Язык C#. Базовый курс [Электронный ресурс] : учеб. пособие / В.В. Подбельский. - 2-е изд., перераб. и доп. - М. : Финансы и статистика, 2015. - <http://www.studentlibrary.ru/book/ISBN9785279035342.html>
2. Информатика 2015 [Электронный ресурс] : учебное пособие / Алексеев А.П. - М. : СОЛОН-ПРЕСС, 2015. - <http://www.studentlibrary.ru/book/ISBN9785913591586.html>
3. Разработка Паскаль-компилятора [Электронный ресурс] / Залогова Л.А. - 3-е изд. (эл.). - М. : БИНОМ, 2014. - <http://www.studentlibrary.ru/book/ISBN9785996325269.html>

7.2. Дополнительная литература

1. "Построение компиляторов [Электронный ресурс] / Никлаус Вирт ; Пер. с англ. Борисов Е. В., Чернышов Л. Н. - М. : ДМК Пресс, 2010." - <http://www.studentlibrary.ru/book/ISBN9785940745853.html>
2. Объектно-ориентированное программирование в C++: лекции и упражнения [Электронный ресурс] : Учебное пособие для вузов / Ашарина И.В. - М. : Горячая линия - Телеком, 2012. - <http://www.studentlibrary.ru/book/ISBN9785991270014.html>
3. Системное и прикладное программное обеспечение [Электронный ресурс] : Учебное пособие / Иванова Н.Ю., Маняхина В.Г. - М. : Прометей, 2011. - <http://www.studentlibrary.ru/book/ISBN9785426300781.html>
4. Основы программирования в ядре операционной системы GNU/Linux [Электронный ресурс] : Учеб. пособие / В.А. Крищенко, Н.Ю. Рязанова. - М. : Издательство МГТУ им. Н. Э. Баумана, 2010. - http://www.studentlibrary.ru/book/bauman_0335.html
5. Теория и реализация языков программирования [Электронный ресурс] / Серебряков В.А. - М. : ФИЗМАТЛИТ, 2012. - <http://www.studentlibrary.ru/book/ISBN9785922114172.html>

7.3. Интернет-ресурсы

1. Электронная библиотека www.citforum.ru

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

«Системное программное обеспечение»

8.1. Программное и коммуникационное обеспечение

Операционная система Windows, стандартные офисные программы MS Office, среда разработки Visual Studio, Интернет-ресурсы.

8.2. Электронные средства обучения

Набор слайдов, методические указания к выполнению лабораторных и практических работ, к курсовому проекту, учебная цифровая вычислительная машина, контрольные тесты.

8.3. Лабораторное оборудование

Лабораторные занятия проводятся в компьютерной лаборатории с использованием персональных компьютеров с установленной лицензионной средой разработки Visual Studio

При проведении лабораторных работ используется мультимедиа проектор и интерактивная доска.

8.4. Средства вычислительной техники и демонстрационное оборудование

Лекции читаются в аудитории кафедры ВТ, оснащенной мультимедиа проектором. При выполнении самостоятельной работы по освоению дисциплины студенты имеют возможность работать в компьютерном классе кафедры ВТ с выходом в сеть Интернет, используя лицензионное прикладное и системное программное обеспечение, а также электронные методические материалы.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению 09.03.01 – «Информатика и вычислительная техника»

Рабочую программу составил ст. преподаватель каф. ВТ Маскеев С.В. 

Рецензент (ы) к.т.н., доцент кафедры ВТ К.В. Куликов 
начальник отдела интеграции ПО ЗАО ИТФ «Системы и технологии»
 А.В. Игнатичев

Программа рассмотрена и одобрена на заседании кафедры ВТ
протокол № 6 от 15 февраля 2016 года.

Заведующий кафедрой  В.Н. Ланцов

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии направления
протокол № 1 от 15 февраля 2016 года.

Председатель комиссии  В.Н. Ланцов

ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Рабочая программа одобрена на 2016/17 учебный год

Протокол заседания кафедры № 10 от 30.08.16 года

Заведующий кафедрой _____

Рабочая программа одобрена на 2017/18 учебный год

Протокол заседания кафедры № 1 от 6.09.17 года

Заведующий кафедрой _____

Рабочая программа одобрена на 2018/19 учебный год

Протокол заседания кафедры № 1 от 14.09.18 года

Заведующий кафедрой _____

Рабочая программа одобрена на 2019/20 учебный год

Протокол заседания кафедры № 1 от 10.09.19 года

Заведующий кафедрой _____

Рабочая программа одобрена на 2020/2021 учебный год

Протокол заседания кафедры № 1 от 31.08.20 года

Заведующий кафедрой _____