

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



Проректор  
по учебно-методической работе

А.А.Панфилов

« 06 » 04 2015 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**«Кроссплатформенное программирование»**

Направление подготовки 09.03.04 "Программная инженерия"

Профиль подготовки "Разработка программно-информационных систем"

Уровень высшего образования бакалавриат

Форма обучения очная

Семестр	Трудоем- кость зач. ед, час.	Лек- ции, час.	Практич. занятия, час.	Лаборат. работы, час.	СРС, час.	Форма промежуточного контроля (экз./зачет)
4	144/4	36		36	36	Экзамен (36)
Итого	144/4	36		36	36	Экзамен (36)

Владимир, 2015 г.

*Handwritten signature*

## 1. Цели освоения дисциплины

Целью преподавания дисциплины является формирование системного базового представления, по основам кроссплатформенного программирования на примере платформы Java и навыков создания приложений в интегрированных средах разработки.

Задачи: изучение теоретических основ о платформе JavaSE и о принципах разработки приложений на этой платформе, научиться пользоваться документацией.

## 2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Кроссплатформенное программирование» относится к обязательной дисциплине вариативной части блока Б1.

Для изучения дисциплины «Кроссплатформенное программирование» являются: знание основных положений математической логики, теории автоматов и формальных языков, теории алгоритмов и технологии программирования; умение использовать технические средства ЭВМ и систем в составе систем обработки информации и управления; обладание опытом работы на ЭВМ с пакетами прикладных программ и владение основными методами, способами и средствами получения, хранения, переработки информации.

Для успешного освоения дисциплины необходимы знания по следующим дисциплинам и разделам ОП: – Алгоритмизация и программирование. – Языки программирования.

Дисциплины для которых знания полученные в результате освоение данной дисциплины являются: «Конструирование программного обеспечения», «Проектирование и архитектура программных систем», «Информационные технологии» «Распределенные программные системы»

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В процессе освоения дисциплины студент формирует и демонстрирует следующие профессиональные компетенции (ПК,19,21):

ПК-19 - владением навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения

ПК-21 - владением навыками чтения, понимания и выделения главной идеи прочитанного исходного кода, документации

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

**Знать:** современное состояние и принципиальные возможности языка программирования Java; концепции кроссплатформенного программного обеспечения; характеристики существующих платформ разработки и области их применения (ПК,19,21).

**Уметь:** использовать полученные знания для создания прикладных программ на языке Java в различных предметных областях(ПК,19,21).

**Владеть:** приемами разработки прикладных программ на языке Java, навыками работы в интегрированных средах разработки; методами отладки и профилирования создаваемых приложений (ПК,19,21).

#### 4. Структура и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 4 зачетных единицы, 144 часа.

№ п/п	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Объем учебной работы с примене нием интеракт ивных методов (в часах / % аудитор ных занятий	Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции	Лабораторные работы	Практические занятия	СРС	КП/КР		
1	2	3	4	5	6	7	8	9	10	11
1	Содержание курса. Цели и задачи дисциплины. История появления компьютеров. Переход от машинных языков к высокоуровневым..	4	1-2	4	4		4		4 / 50	Рейтинг-контроль №1 (05,06 недели)
2	LISP как первый язык с потенциальной поддержкой кроссплатформенно сти.	4	3-4	4	4		4		4 / 50	
3	Портируемые программы. Концепция потоков ввода/вывода. Поддержка обратной совместимости.	4	5-6	4	4		4		4 / 50	
4	Появление промежуточных кодов P-code, как предвестник языка Java.	4	7-8	4	4		4		4 / 50	
5	Язык Java - история развития.. Базовые конструкции языка Java. Переменные и их типы.	4	9-10	4	4		4		4 / 50	
6	Операторы ветвления. Операторы циклов..	4	11-12	4	4		4		4 / 50	

	Вызовы процедур и функций.									
7	Классы и концепция ООП, Полиморфизм, абстрагирование, инкапсуляция.	4	13-14	4	4		4		4 / 50	Рейтинг-контроль №3 (17,18 неделя)
8	Работа со строками. Обработка исключений. Управление памятью.	4	15-16	4	4		4		4 / 50	
9	Коллекции в Java.	4	17-18	4	4		4		4 / 50	
	Всего:			36	36		36		36 / 50	Экзамен

## 5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В рамках дисциплины предусматриваются занятия, проводимые с использованием компьютерных образовательных технологий. При этом в Системе электронного обучения размещаются:

- рабочая программа дисциплины;
- план изучения дисциплины;
- теоретический курс;
- тестирование по теоретическому курсу;
- лабораторные работы:
- методические указания к выполнению лабораторных работ;
- задания к лабораторным работам - индивидуальные варианты;
- вопросы к промежуточному контролю;
- форум общего доступа;
- индивидуальное консультирование.

Контрольные мероприятия при проведении занятий с применением компьютерных образовательных технологий: тестирование в системе электронного обучения по всем разделам дисциплины, проверка выполненных заданий к лабораторным работам, заданий на самостоятельную работу, проведение рейтинг-контроля.

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

По дисциплине предусмотрен текущий контроль в форме рейтинг-контроля и промежуточная аттестация – экзамен.

### Содержание лабораторных работ.

1. Начало работы с Java SE
2. Классы и объекты. Пакеты.
3. Интерфейсы. Наследование.
4. Фреймворк коллекций.
5. Списочные структуры.

6. Деревья и их реализация в языке Java.
7. Деревья-списками узлов.
8. Автоматизированное тестирование программных модулей.

### **Примерный перечень вопросов для текущего контроля:**

#### Рейтинг-контроль 1

1. Дайте определения полиморфизма. Какие особенности его реализации на языке Java?
2. Назовите отличия индустриального и экстремального программирования. Какие приёмы программирования существуют для двух этих подходов в языке Java
3. Опишите основные отличия класса String от StringBuffer?
4. Что такое механизм автоматической сборки мусора (garbage collector)?
5. Основные классы библиотеки для разработки тестов JUNIT
6. Какие действия необходимо произвести, чтобы создать компонент и подписаться на событие, которое он генерирует?
7. Из каких частей состоит заголовок объявления класса? Тело класса?
8. Какие элементы языка Java имеют имена? Какие из них должны быть объявлены? Какие элементы языка Java имеют имена? Какие из них должны быть объявлены?
9. Сравните использование операторов выбора switch в языке Pascal и Java.
10. Как отсортировать массив пользовательских объектов на языке Java ?
11. Когда для реализации списка типа List нужно использовать объект класса ArrayList, а когда LinkedList ?
12. Какие особенности использования коллекций для списков, множеств, пар значений?
13. Какие изменения для работы с коллекциями были внесены в версии Java 2 SE 5.0?
14. Ввести число, занести его цифры в стек. Вывести в число, у которого цифры идут в обратном порядке.
15. Задать два стека, поменять информацию местами.
16. Для каких элементов класса работает полиморфизм?
17. Нарисуйте иерархию классов ввода вывода на основе классов InputStream OutputStream.
18. Какие модификаторы позволяют обращаться к элементу из классов того же пакета?
19. Как генерируется документация утилитой javadoc и набор воспринимаемой утилитой тегов в тексте программы для формирования документации, если в классе заводится новый элемент, и пока нет никаких факторов, позволяющих выбрать тот или иной модификатор доступа. Какой модификатор использовать в таком случае?
20. Есть ли какие-либо ограничения на набор интерфейсов, которые может реализовывать класс?

#### Рейтинг-контроль 2

1. Нарисуйте иерархию классов ввода вывода на основе классов Reader и Writer ?
2. Как реализованы в классе Object методы equals(), toString(), hashCode()?
3. Какие объекты могут быть клонированы?
4. Какие условия должны быть выполнены при переопределении метода equals()?
5. Сколько объектов может быть создано в процессе выполнения клонирования одного объекта средствами JVM?
6. Хотя примитивные массивы не могут участвовать в преобразованиях, однако массивы int[][] и byte[][] могут рассматриваться как одномерные объектные массивы, основанные на ссылочном типе «одномерный примитивный массив». Могут ли такие типы быть преобразованы из одного в другой?
7. Как определить, можно ли преобразовать один тип массива к другому?
8. Сколько объектов порождается при инициализации массива new int[3][4]? new int[3][][ ]?

9. Для каких примитивных типов Java существуют классы-обертки? Что будет получено в результате выполнения: `(new Integer(1)).equals(new Byte(1))` ?
10. Чем отличается внутренний (inner) класс от вложенного (nested) класса?
11. Может ли переменная иметь тип абстрактный класс? Интерфейс? Если да, то какие значения она может хранить?
12. Может ли переменная иметь тип абстрактный класс? Интерфейс? Если да, то какие значения она может хранить?
13. Как узнать, какие события генерирует стандартный графический компонент?
14. Опишите жизненный цикл потока.
15. Если объявить метод `synchronized`, то какой эффект будет этим достигнут?
16. Опишите механизмы синхронизации потоков
17. Как работают `static synchronized` методы?
18. Если один поток начал исполнение `synchronized`-блока, указав ссылку на некий объект, может ли другой поток обратиться к полю этого объекта? К методу?
19. Почему метод `wait` требует обработки `InterruptedException`, а методы `notify` и `notifyAll` – нет?
20. Может ли поток никогда не выйти из метода `wait`, даже если будет вызван метод `notify` или `notifyAll`?

### Рейтинг-контроль 3

1. Какие действия необходимо предпринять для установления TCP соединения между
2. Какие действия необходимо предпринять для установления TCP соединения между двумя Java-приложениями?
3. Какие действия необходимо предпринять для обмена данными по UDP протоколу?
4. Можно ли с помощью класса `URL` пересылать данные на сервер?
5. Для чего нужны менеджеры компоновки? Исходя из каких параметров они выполняют свою работу?
6. Напишите класс-компоненту, у которого по центру рисуется квадрат размерами 10x10.
7. Может ли быть дважды вызван метод `init` у апплета? Метод `start`?
8. Опишите на память иерархию классов, применяемых для построения визуальных приложений
9. Промоделировать вращение спутника вокруг планеты по эллиптической орбите. Когда скрывается за планетой – спутник не виден.
10. Создать фрейм и разместить на нем окружность (одну или несколько). Объект должен «убегать» от указателя мыши. При приближении на некоторое расстояние объект появляется в другом месте фрейма.
11. Имея два объекта класса `File`, каким образом будет наиболее корректно узнать, указывают ли они на одну и ту же директорию (и на директорию ли)? Возможно ли только с помощью этих двух объектов удалить директорию? Если да, то как изменится содержимое другого объекта (если они действительно указывают на одну и ту же директорию)?
12. Какие классы предоставляют методы для записи в поток двоичного представления значений примитивных типов Java?
13. Если необходимо записать (и после считать) несколько строк в файл (из файла), в каком порядке и какие следует настроить фильтры (и для чтения, и для записи)? Какие из них можно пропустить?
14. Что произойдет при попытке к одному объекту `PipedWriter` присоединить несколько различных объектов `PipedReader`? Что произойдет, если несколько раз подряд присоединять один и тот же `PipedReader`?
15. Какая кодировка используется классом `OutputStreamWriter` по умолчанию?

16. Что будет записано в поток, если вызвать метод print() класса PrintWriter, передав в качестве параметра new File("d:\word.txt") ?
17. Какие значения могут быть переданы в конструктор RandomAccessFile для указания режима доступа (чтение/запись)?
18. Какое значение следует передать методу seek() объекта RandomAccessFile, чтобы последний байт файла был считан одиночным вызовом read()?
19. Какие методы объявлены в интерфейсе Serializable?
20. Что произойдет, если записать в файл, используя ObjectOutputStream, значения типов long, int, byte именно в таком порядке, а считать в обратном, используя DataInputStream?

### **Примерный перечень вопросов к экзамену**

1. История создания языка.
2. Основные направления Java.
3. JDK и JRE.
4. Среды разработки для Java.
5. Синтаксис языка.
6. Типы данных.
7. Операторы.
8. Управляющие конструкции.
9. Массивы.
10. Абстракция.
11. Объекты.
12. Абстрактные классы.
13. Интерфейсы.
14. Управление доступом.
15. Инкапсуляция.
16. Наследование и полиморфизм.
17. Коллекции объектов.
18. Обработка ошибок и исключения.
19. Внутренние и анонимные (безымянные) внутренние классы.
20. Система ввода-вывода Java.
21. Интерфейсы Observable, Iterable, Comparable, Cloneable
22. События и их слушатели (ActionListeners)
23. Библиотека Swing.
24. Model-View-Controller.
25. Диспетчеры компоновки.
26. Библиотека SWT.
27. Работа с сетевыми протоколами.
28. Интернационализация.
29. Работа с базами данных.
30. Работа со звуком и графикой.

### **Примерный перечень тем для самостоятельного изучения**

Рекомендуемая тематика рефератов по курсу:

1. Кроссплатформенные фреймворки: достоинства и недостатки. Сравнительная характеристика.
2. Кроссплатформенные фреймворки: цены и тарифные планы.
3. Конфигурирование сборки проектов в зависимости от платформы.
4. Системы контроля версий: общий обзор и сравнительная характеристика. 5. Клиент-серверные и распределенные системы контроля версий.

6. Обзор инструментальных средств разработки кроссплатформенных приложений с использованием библиотеки Qt.
7. Кроссплатформенные приложения с использованием библиотеки Juce.
8. Кроссплатформенные приложения на языке Java: отличительные особенности, преимущества и недостатки.
9. Кроссплатформенные приложения на языке Java: инструментальные среды разработки.
10. Кроссплатформенные приложения на языке Python: отличительные особенности, преимущества и недостатки.
11. Кроссплатформенные приложения на языке Python: инструментальные среды разработки.
12. Апплеты и сервлеты.
13. Разработка приложений под мобильные устройства.

## **7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### ***а) основная литература:***

1. Шутов А. В. Медведев Ю. А. Структуры и алгоритмы компьютерной обработки данных: [в ч.]. Ч. 2: Лабораторный практикум 2013
2. Python и анализ данных [Электронный ресурс] / Уэс Маккинли - М. : ДМК Пресс, 2015.
3. Объектно Ориентированное Программирование. Хорошая книга для Хороших Людей [Электронный ресурс] / Комлев Н.Ю. - М. : СОЛОН-ПРЕСС, 2015. - <http://www.studentlibrary.ru/book/ISBN9785913591388.html>

### ***б) дополнительная литература:***

1. В.В. Вершинин Программирование для Microsoft.NET : методические указания к лабораторным работам по дисциплине "Программирование" : в 2 ч. / В. В. Вершинин, С. В. Чебыкин ; Владимирский государственный университет (ВлГУ) 2010
2. Основы проектирования корпоративных систем [Электронный ресурс] / Зыков С.В. - М. : ИД Высшей школы экономики, 2012. - <http://www.studentlibrary.ru/book/ISBN9785759808626.html>
3. Андрианов Д. П. Методические указания к лабораторным работам по дисциплине "Системное программирование" 2010.

### ***в) периодические издания:***

1. Вестник компьютерных и информационных технологий ISSN 1810-7206.

### ***г) интернет-ресурсы***

- [www.edu.ru](http://www.edu.ru) – портал российского образования
- [www.elbib.ru](http://www.elbib.ru) – портал российских электронных библиотек
- [www.distance-learning.ru](http://www.distance-learning.ru) – портал, посвященный дистанционному обучению
- [www.eLibrary.ru](http://www.eLibrary.ru) – научная электронная библиотека
- [www.moodle.com](http://www.moodle.com) – портал разработчиков Moodle
- [library.vlsu.ru](http://library.vlsu.ru) - научная библиотека ВлГУ
- [www.cs.vlsu.ru:81/ikg](http://www.cs.vlsu.ru:81/ikg) – учебный сайт кафедры ИСПИ ВлГУ
- <https://vlsu.bibliotech.ru/> - электронная библиотечная система ВлГУ

### ***в) периодические издания:***



1. Вестник компьютерных и информационных технологий ISSN 1810-7206.

2) интернет-ресурсы

- www.edu.ru – портал российского образования
- www.elbib.ru – портал российских электронных библиотек
- www.distance-learning.ru – портал, посвященный дистанционному обучению
- www.eLibrary.ru – научная электронная библиотека
- www.moodle.com – портал разработчиков Moodle
- library.vlsu.ru - научная библиотека ВлГУ
- www.cs.vlsu.ru:81/ikg – учебный сайт кафедры ИСПИ ВлГУ
- https://vlsu.bibliotech.ru/ - электронная библиотечная система ВлГУ

**8. Материально-техническое обеспечение дисциплины**

1. Лабораторные работы проводятся в компьютерном классе кафедры ИСПИ (лаб. 314-3; 13 компьютеров) с использованием установленного программного обеспечения.
2. Лекции читаются в мультимедийных аудиториях кафедры ИСПИ, оборудованных электронными проекторами (ауд. 314-3; 213-3), с использованием комплекта слайдов. Компьютерный класс оснащенный современными компьютерами, объединенными локальными вычислительными сетями с выходом в Интернет.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.04 «Программная инженерия» Профиль подготовки «Разработка программно-информационных систем»

Рабочую программу составили доцент кафедры Озерова М.И.

ст. пр. Шевченко Д.В.

Рецензент: начальник отдела Системной и технической поддержки вычислительного комплекса ГУ БР по Владимирской области, к.т.н. А.Г. Долинин

Программа рассмотрена и одобрена на заседании кафедры ИСПИ

протокол № 711 от 06.04.15 года.

Заведующий кафедрой

Жигалов И.Е.

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии направления 09.03.04 «Программная инженерия»

протокол № 7 от 06.04.15 года.

Председатель комиссии

Жигалов И.Е.