

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет математики и компьютерных наук

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Java – программирование интернет приложений

Кафедра дискретной математики и информатики
факультета математики и компьютерных наук

Образовательная программа
02.04.02 Фундаментальная информатика и информационные
технологии

Направленность (профиль) подготовки:
Информационные технологии

Уровень высшего образования
магистратура

Форма обучения
Очная-заочная

Статус дисциплины:
входит в часть ОПОП, формируемую участниками
образовательных отношений

Махачкала, 2021

Рабочая программа дисциплины «Java – программирование интернет приложений» составлена в 2021 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.04.02 - Фундаментальная информатика и информационные технологии.

Приказ Министерства образования и науки РФ от 23 августа 2017 г. N 811 "Об утверждении федерального государственного образовательного стандарта высшего образования - магистратура по направлению подготовки 02.04.02 Фундаментальная информатика и информационные технологии" (с изменениями и дополнениями). Редакция с изменениями N 1456 от 26.11.2020

Разработчик: кафедра дискретной математики и информатики, Якубов А.З., к. ф.-м. н., доцент.

Рабочая программа дисциплины одобрена:
на заседании кафедры дискретной математики и информатики

от 30.05.2021 г., протокол № 9.

Зав. кафедрой  Магомедов А.М.

и
на заседании Методической комиссии факультета математики и компьютерных наук от 23.06.2021 г., протокол № 6.

Председатель  Бейбалаев В.Д.

Рабочая программа дисциплины согласована с учебно-методическим управлением « 09 » 07 2021 г.

Начальник УМУ  Гасангаджиева А.Г.

Аннотация рабочей программы дисциплины

Дисциплина “Java-программирование интернет приложений” входит в часть ОПОП, формируемую участниками образовательных отношений программы магистратуры по направлению 02.04.02 - Фундаментальная информатика и информационные технологии.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с объектно-ориентированным программированием, разработкой Web-приложений, созданием консольных и GUI-приложений.

Дисциплина нацелена на формирование следующих компетенций выпускника: ПК-1 ПК-4.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: практических и лабораторных занятий.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: в форме 2-х контрольных работ в конце 3 и 5 модуля и итогового зачета в конце семестра.

Объем дисциплины – 6 зачетных единиц, в том числе в академических часах по видам учебных занятий:

Се- местр	Учебные занятия						СРС	Форма промежуточной аттестации экзамен
	в том числе							
	Контактная работа обучающихся с преподавателем							
	Все го	из них						
Лек- ции		Лабораторные занятия	Практические занятия	КСР	консультации			
2	216		12	12			156	36

1. Цели освоения дисциплины

Целями освоения дисциплины Java-программирование интернет приложений являются овладение знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования. Формирование у студентов научного, творческого подхода к освоению технологий, методов и средств производства программного обеспечения для глобальной сети интернет.

Основные задачи дисциплины: овладение методами структурного и объектно-ориентированного программирования; закрепление навыков алгоритмизации и программирования, получение знаний и навыков, позволяющего выпускнику успешно работать в избранной сфере деятельности с применением современных компьютерных технологий.

Ожидаемые результаты:

- получить знания об особенностях объектно-ориентированного программирования на Java;
- изучить средства и методы, предоставляемые языком Java для Web-программирования;
- освоить современные методы программирования апплетов, сервлетов, способы их взаимодействия в сети;
- получить необходимые знания о протоколах передачи информации глобальных сетей;
- приобрести навыки и умения в постановке и решении задач разработки динамических Web-страниц.

2. Место дисциплины в структуре ОПОП магистратуры

Дисциплина входит в вариативную часть образовательной программы магистратуры по направлению 02.03.02 - Фундаментальная информатика и информационные технологии и изучается в соответствии с графиком учебного процесса в десятом семестре. Изучение предмета производится в течение одного семестра и заканчивается экзаменом.

Дисциплина опирается на знания, полученные в 1 семестре в процессе изучения дисциплин «Основы программирования», во 2 семестре «Языки программирования», в 3 семестре «Программирование на основе классов», в 4 семестре «Основы Web-программирования». В свою очередь, знания, умения и навыки, полученные при изучении данной дисциплины, используются, закрепляются и развиваются при проведении преддипломной практики, выполнении магистерской диссертации.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения)

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций	Планируемые результаты обучения	Процедура освоения
ПК-1. Способность понимать и применять в научно-исследовательской и прикладной деятельности современный математический аппарат, ос-	ПК-1.1. Знает основы ведения научной дискуссии и формы устного научного высказывания.	<i>Знает:</i> базовые основы современного математического аппарата, связанного с проектированием программных продуктов и программных комплексов, ос-	Конспектирование и проработка теоретического материала. Участие в практических занятиях. Реализация проектов на лабораторных занятиях. Самостоятель-

<p>новные законы естествознания, современные языки программирования и программное обеспечение; операционные системы и сетевые технологии.</p>		<p>новы дискретной математики и математической логики, теории вероятностей, математической статистики и случайных процессов, численных методов. <i>Умеет:</i> использовать математический аппарат в профессиональной деятельности. <i>Владеет:</i> практическим опытом применения современного математического аппарата, связанного с проектированием и разработкой программных продуктов и программных комплексов.</p>	<p>ная работа.</p>
	<p>ПК-1.2. Умеет вести корректную дискуссию в области информационных технологий, задавать вопросы и отвечать на поставленные вопросы по теме научной работы.</p>	<p><i>Знает:</i> базовые понятия в области математических наук и программирования. <i>Умеет:</i> находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности. <i>Владеет:</i> практическим опытом научно-исследовательской деятельности в математике и информатике.</p>	
	<p>ПК-1.3. Имеет практический опыт владения существующими методами и алгоритмами решения задач цифровой обработки сигналов, использования сети Интернет, аннотирования, реферирова-</p>	<p><i>Знает:</i> основные методы проектирования и производства программного продукта. <i>Умеет:</i> использовать методы проектирования и производства программ-</p>	

	<p>ния, библиографического разыскания и описания, опыт работы с научными источниками.</p>	<p>ного продукта. <i>Владеет:</i> практическим опытом применения указанных выше методов и технологий.</p>	
<p>ПК-4. Способность применять в профессиональной деятельности современные языки программирования и методы параллельной обработки данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии.</p>	<p>ПК-4.1. Знает современные языки программирования и методы параллельной обработки данных.</p>	<p><i>Знает:</i> основные методы решения прикладных задач, современные методы информационных технологий, языки программирования и современные информационные технологии. <i>Умеет:</i> корректно оформить результаты научного труда в соответствии с современными требованиями; составлять программы на современных языках программирования. <i>Владеет:</i> Имеет практический опыт использования сети Интернет, аннотирования, реферирования, библиографического разыскания и описания, опыт работы с научными источниками; навыками программирования на современных языках</p>	<p>Конспектирование и проработка теоретического материала. Участие в практических занятиях. Реализация проектов на лабораторных занятиях. Самостоятельная работа.</p>
	<p>ПК-4.2. Умеет реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии</p>	<p><i>Знает:</i> Знает методику установки и администрирования программных систем.; различные языки программирования. <i>Умеет:</i> реализовывать техническое сопровождение информационных систем; применять различные языки программирования в</p>	

		<p>численном анализе. <i>Владеет:</i> Имеет практический опыт разработки интеграции информационных систем с использованием аппаратно-программных комплексов</p>	
	<p>ПК-4.3. Имеет практический опыт разработки интеграции информационных систем.</p>	<p><i>Знает:</i> современные языки программирования и методы параллельной обработки данных. Знаком с содержанием Единого Реестра Российских программ для электронных вычислительных машин и баз данных; современные информационные технологии. <i>Умеет:</i> реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии, с использованием современных информационных технологий. <i>Владеет:</i> навыками решения задач с использованием различных информационных технологий, имеет практический опыт разработки интеграции информационных систем.</p>	

4. Объем, структура и содержание дисциплины

4.1. Объем дисциплины составляет 6 зачетных единиц, 216 академических часов: 12ч. практических занятий, 12 ч. Лабораторных занятий, 156 – СРС, 36ч. -экзамен.

4.2. Структура дисциплины

Структура и содержание дисциплины «Java-программирование интернет приложений»

№ п/п	Разделы и темы дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Самостоятельная работа	Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
			Лекции	Практические занятия	Лабораторные занятия	Контроль самостоятел. раб.		
Модуль 1. Основы языка Java.								
1	Введение в программирование на Java. Инструментальные средства разработки Java-программ.	А		1	1		8	Текущий контроль - тест 1.1
2	Базовые элементы языка. Типы.	А		1	1		10	Текущий контроль - тест 2.1
3	Базовые элементы языка. Операторы.	А		1	1		18	Текущий контроль - тест 3.1
	<i>Итого по модулю 1: 36</i>			3	3		36	Устный опрос
Модуль 2. Классы и объекты.								
1	Основы объектно-ориентированного программирования для Java. Классы и объекты.	А		1	1		10	Текущий контроль - тест 4.1
2	Работа со строками.	А		1	1		8	Текущий контроль - тест 5.1
3	Обработка исключений на Java.	А		1	1		10	Текущий контроль - тест 6.1
	<i>Итого по модулю 2: 36</i>			3	3		28	Устный опрос
Модуль 3. Интернет-программирование, апплеты.								
1	Создание апплета.	А		1	1		12	Текущий контроль - тест 1.2

2	Компиляция и выполнение апплета. Передача параметров в апплет.	A			1	1		10	Текущий контроль - тест 2.2
3	Средства пользовательского интерфейса. Компоненты и контейнеры.	A			1	1		10	Текущий контроль - тест 3.2
	<i>Итого по модулю 3: 36</i>				3	3		32	Контрольная работа №1
Модуль 4. Обработка событий от компонент.									
1	События. Обработка событий от компонент.	A			1	1		14	Текущий контроль - тест 4.2
2	Растровые изображения и анимация в апплетах. Звук в апплетах Java.	A			1	1		20	Текущий контроль - тест 5.2
	<i>Итого по модулю 3: 36</i>				2	2		34	<i>Контрольная работа №1</i>
Модуль 5. Создание потоков.									
1	Программирование меню на Java. Создание потоков связанных с файлами. Файловый ввод/вывод.	A			1	1		26	Текущий контроль - тест 6.2 Текущий контроль - тест 7.2
	<i>Итого по модулю 5: 36</i>				1	1		26	Устный опрос
Модуль 6. Экзамен.									
	<i>Итого по модулю 6: 36</i>								36 Экзамен
	ИТОГО: 216				12	12		156	36

4.3. Содержание дисциплины, структурированное по темам

4.3.1. Содержание практических занятий по дисциплине

Тема 1. Введение в программирование на Java. Инструментальные средства разработки Java-программ.

JDK – инструментальные средства разработки Java-программ.

Документация, литература.

Понятие байт-кода.

Платформено-независимость Java-приложений.

Мобильность Java.

Пакетные средства, интегрированные средства разработки.

Тема 2. Базовые элементы языка. Типы.

Структура программы.

Типы данных.

Идентификаторы, константы, переменные.

Преобразование типов.
Массивы.

Тема 3. Базовые элементы языка. Операторы.

Операторы выбора, цикла, перехода.
Операторы и блоки.
Оператор if-else.
Оператор switch.
Цикл while и do-while.
Оператор for. Метки.
Оператор break.
Оператор continue.
Оператор return.
Использование функций.

Тема 4. Основы объектно-ориентированного программирования для Java.

Классы и объекты.

Основные понятия.
Наследование.
Классы, интерфейсы, абстрактные классы.
Реализация классов.
Использование полиморфизма.
Создание объекта, оператор new.
Правила доступа к данным и методам объекта.
Переопределение методов.
Метод-конструктор.

Тема 5. Работа со строками.

Классы String, StringBuffer.
Доступ к символам.
Равенство строк. Сравнение.
Упорядочение.
Методы работы со строками.
Методы indexOf, lastIndexOf.

Тема 6. Обработка исключений на Java.

Типы исключений.

Вложенные операторы try.
Блок finally.
Операторы throw, catch.
Подклассы Exception.

Тема 7. Создание апплета.

Создание апплета на Java.
Параметры апплета.
Исходный текст апплета.
Методы init, destroy, start, stop.

Тема 8. Компиляция и выполнение апплета. Передача параметров в апплет.

Компиляция и выполнение апплета.
Выполнение апплета в отдельном потоке.
Параметры апплета.
Компоненты в окне апплета.

Тема 9. Средства пользовательского интерфейса. Компоненты и контейнеры.

Кнопки, переключатели.
Списки класса Choice.
Списки класса List.
Текстовое поле классов Label, TextField.
Многострочное текстовое поле класса TextArea.

Тема 10. События. Обработка событий от компонент.

Обработка событий.
Метод `handleEvent`.
Класс `MouseEvent` и интерфейс `MouseListener`.
Движение мыши, перетаскивание.
Событие от клавиатуры.
Обработка событий, инициированных компонентами.

Тема 11. Растровые изображения и анимация в апплетах. Звук в апплетах Java.

Загрузка и рисование растрового изображения.
Класс `Image`. Метод `paint`.
Координаты, цвет, шрифт.
Графические элементы. Класс `Graphics`. Метод `setColor`.
Просмотр изображения в апплете.
Видео в окне апплета.
Загрузка и проигрывание звуковых файлов.

Тема 12. Создание, выполнение и синхронизация потоков.

Многопоточность. Приоритеты потоков.

Процессы, потоки и приоритеты. Создание потоков.
Синхронизация. Методы `synchronized`. Операторы `synchronized`.
Методы `wait` и `notify`. Планирование потоков.
Взаимная блокировка. Приостановка потоков. Прерывание потока.
Завершение работы потока. Завершение приложения.
Использование `Runnable`. Ключевое слово `volatile`.
Безопасность потоков и `Thread Group`. Отладка потоков.
Реализация многопоточности в Java.

Тема 13. Создание потоков связанных с файлами. Файловый ввод/вывод.

Потоки ввода и вывода.
Файловый ввод/вывод.
Построчный и побайтный ввод/вывод.

Тема 14. Программирование меню на Java.

Классы `MenuBar`, `Menu` и `MenuItem`.
Создание меню в окне типа `Frame`.
Отображение меню.

Тема 15. Понятие сервлета.

Класс `HttpServlet`. Структура сервлета.

Создание и инициализация сервлета.
API для работы с сервлетами.
Обработка клиентских запросов.

Тема 16. Архитектура, жизненный цикл, размещение сервлетов.

Многопоточность и сервлеты.

Размещение сервлетов.

Технология передачи файлов из браузера в сервлет.

Доступ к базам данных из сервлета Java.

4.3.2. Содержание лабораторных занятий по дисциплине

Лабораторная работа по теме:

Введение в программирование на Java. Инструментальные средства разработки Java-программ.

1. JDK – инструментальные средства разработки Java-программ.
2. В чем заключается платформу-независимость Java-приложений?
3. Что представляет собой исходный файл на языке Java?
4. Написать программу выводящую на экран строку HelloWorld.

Лабораторная работа по теме:

Базовые элементы языка. Типы, операторы.

1. Описать основные стандартные типы данных в языке Java.
2. Описать операторы выбора, цикла.
3. Написать программу, выводящую на экран месяцы года, время года и количество дней в каждом из них, используя, оператор цикла.

Лабораторная работа по теме:

Основы объектно-ориентированного программирования для Java.

Классы и объекты.

1. Описать реализацию класса.
2. Описать оператор new.
3. Объявление методов.
4. Написать программу использующую разную реализацию методов в классе.

Лабораторная работа по теме:

Работа со строками.

1. Описание класса String.
2. Описание класса StringBuffer. Примеры программ.

Лабораторная работа по теме:

Обработка исключений на Java.

1. Описать типы исключений.
2. Описать работу операторов try, throw, catch, finally.
3. Написать программу использующую операторы исключения для выхода из метода.

Лабораторная работа по теме:

Создание Java-приложения. Компиляция и выполнение апплета. Передача параметров в апплет.

1. Описать строение апплета.

2. Компиляция и выполнение апплета на Java.
3. Создать апплет с получением параметров.

Лабораторная работа по теме:

Средства пользовательского интерфейса. Компоненты и контейнеры.

1. Вспомнить определения компонентов и контейнеров.
2. Написать апплеты, демонстрирующие примеры работы с классами Label, Button, Choise, List, TextField, TextArea.

Лабораторная работа по теме:

События. Обработка событий от компонентов.

1. Описать процесс обработки событий от компонентов.
2. Написать программу демонстрирующую работу обработчиков событий.

Лабораторная работа по теме:

Растровые изображения и анимация в апплетах.

Звук в апплетах Java.

1. Написать апплет, выполняющий загрузку изображения.
2. Написать апплет, выполняющий вывод анимации.
3. Написать программу воспроизводящую аудио файл.

Лабораторная работа по теме:

Создание, выполнение и синхронизация потоков. Многопоточность. Приоритеты потоков.

1. Описать класс Thread, на примере управления выполняющимся в данный момент подпроцессом.
2. Написать программу создающую два подпроцесса.
3. Написать программу иллюстрирующую работу оператора synchronized.

Лабораторная работа по теме:

Создание потоков связанных с файлами. Файловый ввод/вывод.

1. Написать программу, считывающую данные с файла, используя класс FileInputStream.
2. Написать программу, иллюстрирующую работу с записью данных в файлы, используя класс FileOutputStream.
3. Описать классы для работы с побайтным вводом/выводом.

Лабораторная работа по теме

Программирование меню на Java.

Написать апплет создающий стандартное меню.

Лабораторная работа по теме:

Понятие сервлета. Архитектура, жизненный цикл, размещение сервлетов

1. Составить описание сервлетов.
2. Создать и выполнить сервлет.

Лабораторная работа по теме:

Особенности создания сетевых приложений. Класс URL в библиотеке классов Java.

1. Описать адрес IP и класс InetAddress.

2. Описать класс URL и показать работу с ним, на примере программы.
3. Сокеты и работа с ними.

5. Образовательные технологии

Для эффективной реализации целей и задач ФГОС, для претворения компетентностного подхода в преподавании дисциплины «Java программирование интернет приложений», используются следующие образовательные технологии и методы обучения:

Вид занятия	Технология	Цель	Формы и методы обучения
1	2	3	4
Практические занятия	Технология проблемного обучения	Усвоение теоретических знаний, развитие мышления, формирование профессионального интереса к будущей деятельности	Мультимедийные лекции-объяснение, лекция-визуализация, с привлечением формы тематической дискуссии, беседы, анализа конкретных ситуаций
Лабораторные занятия	Технология проблемного, модульного, дифференцированного и активного обучения, деловая игра	Развитие творческой и познавательной самостоятельности, обеспечение индивидуального подхода с учетом базовой подготовки. Организация активности студентов, обеспечение лично-деятельного характера усвоения знаний, приобретения навыков, умений.	Индивидуальный темп обучения. Инновационные интерактивные методы в обучении: использование Web-ресурсов для подготовки компьютерных презентаций, использование off-line (электронная почта) для обмена информацией, консультаций с преподавателем, работа с электронными пособиями, возможность самотестирования. Постановка проблемных познавательных задач. Методы активного обучения: «круглый стол», игровое производственное
Самостоятельная работа	Технологии концентрированного, модульного, дифференцированного обучения	Развитие познавательной самостоятельности, обеспечение гибкости обучения, развитие навыков работы с различными источниками информации, развитие умений, творческих навыков	Индивидуальные, групповые, интерактивные (в режимах on-line и offline).

6. Учебно-методическое обеспечение самостоятельной работы студентов

6.1. Примерное распределение времени самостоятельной работы студентов

Раздел	Вид самостоятельной работы - практиче-	Контрольные	Уч.-мет.
--------	----------------------------------------	-------------	----------

(модуль, тема)	содержание	сроки (в нед.) и вид контроля	обеспечение (указаны источники из списка основной литературы)
1	2	3	4
Модуль 1. Основы языка Java. Модуль 2. Классы и объекты. Модуль 3. Интернет-программирование, апплеты.	Введение в программирование на Java. Инструментальные средства разработки Java-программ. Базовые элементы языка. Типы. Базовые элементы языка. Операторы. Основы объектно-ориентированного программирования для Java. Классы и объекты. Работа со строками. Обработка исключений на Java. Создание апплета. Компиляция и выполнение апплета. Передача параметров в апплет. Средства пользовательского интерфейса. Компоненты и контейнеры.	3 и 7 недели обучения. Проверка теоретических знаний на устном опросе и коллоквиуме. Проверка решенных задач.	[1] – [5]; материалы сайтов: https://ru.wikipedia.org/wiki/Java http://citforum.ru/
Модуль 4. Обработка событий от компонент. Модуль 5. Экзамен	События. Обработка событий от компонент. Растровые изображения и анимация в апплетах. Звук в апплетах Java. Создание, выполнение и синхронизация потоков. Многопоточность. Приоритеты потоков. Создание потоков связанных с файлами. Файловый ввод/вывод. Программирование меню на Java. Понятие сервлета. Архитектура, жизненный цикл, размещение сервлетов. Подготовка к экзамену и сдача экзамена	7 и 10 недели обучения. Проверка письменных контрольных работ Проверка выполнения компьютерных программ	[3], [4]; материалы сайтов: http://www.emanual.ru/

6.2. Виды и порядок выполнения самостоятельной работы

1. Изучение рекомендованной литературы.
2. Подготовка к опросу на практических занятиях
3. Решение задач и упражнений
4. Подготовка к коллоквиуму и контрольным работам
5. Поиск материала на интернет-форумах
6. Подготовка к экзамену

6.3. Порядок контроля:

1. Опрос на лабораторном занятии
2. Проверка выполнения домашних заданий и контрольных работ
3. Коллоквиум
4. Экзамен.

6.4. Примеры индивидуальных вариантов задач для самостоятельного выполнения:

Вариант 1

1. Создайте массив из 15 случайных целых чисел из отрезка $[0;9]$. Выведите массив на экран. Подсчитайте сколько в массиве чётных элементов и выведите это количество на экран на отдельной строке.
2. Создайте класс окружностей на плоскости, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных. Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем). Создайте в классе метод, проверяющий имеют ли две окружности равную площадь. С использованием построенного класса создайте две окружности: одну с центром в $(0;0)$ и радиусом 12, а вторую с центром в $(3;5)$ и радиусом 11. Проверьте с помощью созданного метода равна ли их площадь и если равна, то выведите соответствующее сообщение на экран.

Вариант 2

1. Создайте массив из 8 случайных целых чисел из отрезка $[1;10]$. Выведите массив на экран в строку. Замените каждый элемент с нечётным индексом на ноль. Снова выведите массив на экран на отдельной строке.
2. Создайте класс отрезков на координатной плоскости, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных. Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем). Создайте в классе метод, проверяющий равна ли длина двух отрезков. С использованием построенного класса создайте два отрезка: один от точки $(1;1)$ до точки $(2;2)$ и второй отрезок от точки $(-3;0)$ до точки $(1;1)$. Проверьте с помощью созданного метода равна ли их длина и если равна, то выведите соответствующее сообщение на экран.

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

7.1. Типовые контрольные задания

7.1.1. Темы рефератов:

Введение в программирование на Java. Инструментальные средства разработки Java-программ.

Базовые элементы языка. Типы.

Базовые элементы языка. Операторы.

Основы объектно-ориентированного программирования для Java.

Работа со строками.

Обработка исключений на Java.

Компиляция и выполнение апплета. Передача параметров в апплет.

Средства пользовательского интерфейса. Компоненты и контейнеры.

События. Обработка событий от компонент.

Растровые изображения и анимация в апплетах. Звук в апплетах Java.
 Сетевые приложения на Java
 Создание, выполнение и синхронизация потоков. Многопоточность
 Создание потоков связанных с файлами. Файловый ввод/вывод.
 Программирование меню на Java.
 Понятие сервлета.
 Архитектура, жизненный цикл, размещение сервлетов.
 Технология передачи файлов из браузера в сервлет. Доступ к базам данных из сервлета Java.
 Особенности создания сетевых приложений. Класс URL в библиотеке классов Java.

7.1.2. Примерные упражнения и задания для текущего контроля

Контрольная работа №1

Вариант 1

Задание 1. Дан исходный код программы:

```
classExample{
  publicstaticvoid main(String[]args){
    _____ a =0==8%2;
    if(a ||(a &&false)){
      System.out.println((1+1.5)+"2");
    }else{
      System.out.println(3+"a"+"2"+a);
    }
  }
}
```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2. Что выведется на экран в результате работы представленной выше программы?

Задание 3. В программе имеется объявленная переменная s типа int. Напишите выражение, в результате которого переменной s будет присвоено случайное значение из отрезка [-9;0].

Вариант 3

Задание 1. Дан исходный код программы:

```
classExample{
  publicstaticvoid main(String[]args){
    _____ b =5.0;
```

Вариант 2

Задание 1. Дан исходный код программы:

```
classExample{
  publicstaticvoid main(String[]args){
    _____ a =5>3;
    if(a ||(5+3)*2<16){
      System.out.println(4+"12"+2);
    }else{
      System.out.println(2+12*2);
    }
  }
}
```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2. Что выведется на экран в результате работы представленной выше программы?

Задание 3. В программе имеется объявленная переменная s типа int. Напишите выражение, в результате которого переменной s будет присвоено случайное значение из отрезка [4;12].

Вариант 4

Задание 1. Дан исходный код программы:

```
classExample{
  publicstaticvoid main(String[]args){
    _____ a =-3.0;
```

```

b = b*2;
if(b >10|| b <-10){
    System.out.println("12"+0+4);
}else{
    System.out.println(0+4+"12");
}
}
}
}

```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2. Что выведется на экран в результате работы представленной выше программы?

Задание 3. В программе имеется объявленная переменная s типа int. Напишите выражение, в результате которого переменной s будет присвоено случайное значение из отрезка [-6;2].

```

a =2+ a;
if(3>2||3>2&&false){
    System.out.println("4"+a);
}else{
    System.out.println(-a+"4");
}
}
}
}

```

Какой тип данных из набора boolean, int, double нужно подставить вместо прочерка, чтобы программа компилировалась корректно?

Задание 2. Что выведется на экран в результате работы представленной выше программы?

Задание 3. В программе имеется объявленная переменная s типа int. Напишите выражение, в результате которого переменной s будет присвоено случайное значение из отрезка [-6;8].

Контрольная работа №2

Разработка пользовательских классов

Создайте класс окружностей на плоскости, описав в нём все необходимые свойства, подобрав им понятные имена и правильные типы данных.

Опишите в классе конструктор, позволяющий при создании нового объекта явно задать все его свойства. Если это необходимо, то проверьте допустимость их значений в конструкторе (например, в классе обыкновенных дробей нельзя создавать дробь с нулевым знаменателем).

Создайте в классе метод, проверяющий имеют ли две окружности равную площадь.

С использованием построенного класса создайте две окружности: одну с центром в (0;0) и радиусом 12, а вторую с центром в (3;5) и радиусом 11. Проверьте с помощью созданного метода равна ли их площадь и если равна, то выведите соответствующее сообщение на экран.

7.1.3 Контрольные вопросы

Вопросы к экзамену по курсу «Java-программирование интернет приложений» для студентов 1 курса магистратуры ФМиКН направления ФИИТ, 10 семестр.

1. Особенности языка и платформы Java.

2. Классификация программ по типу исполнения (компилируемые, интерпретируемые, исполняемые на виртуальных машинах). Виртуальная машина Java. JIT-компиляция.
3. Создание простейшей программы на Java, её компиляция в байт-код и запуск.
4. Средства разработки Java-приложений. Интегрированные среды разработки.
5. Встроенные типы данных. Способы задания литералов различных типов.
6. Хранение данных в памяти ЭВМ.
7. Приведение типов (явное и автоматическое). Константы и переменные.
8. Оператор присваивания. Порядок действий (приоритет операторов).
9. Арифметические операторы. Операторы инкремента и декремента.
10. Встроенный класс Math. Псевдослучайные числа.
11. Операторы сравнения и логические операторы.
12. Операторы ветвления. Условный оператор. Минимизация количества проверок.
13. Операторы ветвления. Оператор множественного выбора. Его сравнение с условным оператором.
14. Встроенный класс String. Строковые операции.
15. Стандартные потоки ввода-вывода. Организация ввода и вывода данных. Класс Scanner.
16. Операторы организации циклов. Цикл типа «n раз».
17. Операторы организации циклов. Цикл типа «пока» (с пред- и постпроверкой условия).
18. Массивы. Способы объявления и инициализации массивов. Индексация и размер массива.
19. Массивы. Алгоритмы сортировки.
20. Массивы. Многомерные массивы.
21. Статические методы классов. Методы функционального и процедурного типа.
22. Сигнатура метода. Перегрузка методов.
23. Процедурное программирование. Объектно-ориентированное программирование. Сравнение парадигм.
24. Основные понятия ООП. Объекты и классы. Абстракция данных. Сценарий построения объектно-ориентированной программы.
25. Члены классов. Методы и поля.
26. Специальные методы классов (конструкторы). Конструктор по умолчанию.
27. Модификаторы уровня доступа (default, public, protected, private).
28. основополагающие принципы ООП. Инкапсуляция.
29. основополагающие принципы ООП. Наследование. Управление наследованием.
30. Интерфейсы как средство реализации множественного наследования.
31. основополагающие принципы ООП. Полиморфизм. Средства реализации полиморфизма.
32. Иерархия классов Java. Коренной класс Object и его методы.
33. Исключительные ситуации. Обработка исключительных ситуаций.
34. Приложения с графическим интерфейсом с использованием GUI-пакетов и апплеты.

7.4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Общий результат складывается из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- посещение занятий – 10 баллов,

- выполнение текущих лабораторных заданий – 40 баллов,
 - выполнение домашних (аудиторных) контрольных работ - 50 баллов.
- Промежуточный контроль по дисциплине включает:
- устный опрос - 50 баллов,
 - письменная контрольная работа - 50 баллов.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Патрик Ноутон, Герберт Шилдт. Java™2. Наиболее полное руководство. Санкт-Петербург. “БХВ-Петербург” 2013, 1050 стр.
2. В.Будилов. Интернет-программирование на Java. С.Пб: ВHV-Санкт-Петербург, 2014, 694 стр.
3. Вебер Д. Технология Java в подлиннике. С.Пб: ВHV-Санкт-Петербург, 2012, 1104 стр.
4. Эферган М. JAVA Справочник. С.Пб: Питер, 2013. 448 стр.
5. Мейнджер Д. JAVA: Основы программирования. С.Пб: ВHV-Санкт-Петербург, 2012, 320 стр.
6. Мейсо Б. JAVA ++: Основы программирования. 2014, 400 стр.

Дополнительная литература

1. Крис Джамса Библиотека программиста Java .- Jamsa Press, ООО "Попурри", 2009.
2. И.Ю.Баженова Язык программирования Java .- АО "Диалог-МИФИ", 2011
3. Нейл Бартлетт, Алекс Лесли, Стив Симкин Программирование на Java. Путеводитель .- The Coriolis Group, Inc., 1996, Издательство НИПФ "ДиаСофт Лтд.", 2012

9. Перечень рекомендуемых ресурсов информационно-телекоммуникационной сети «Интернет»

<https://ru.wikipedia.org/wiki/Java>

<http://citforum.ru/>

<http://www.compdoc.ru/>

<http://www.emanual.ru/>

10. Методические указания для обучающихся по освоению дисциплины

- 1) Помимо выполнения лабораторных заданий рекомендуется решить все упражнения (или большую часть) упражнений, предложенных к каждому практическому занятию.

2) Самостоятельная работа студентов заключается в решении всех разобранных на занятиях упражнений, материала учебника и соответствующих форумов интернет, решения всех заданий из индивидуальных лабораторных заданий, решения рекомендуемых задач, подготовки к сдаче промежуточных отчетов и зачета и дополнительной работы в компьютерном классе самостоятельно.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Для проведения полноценных занятий необходимо следующее программное обеспечение: операционная система Windows 7, 8.1 и 10, JDK, библиотеки Java, одна из программных оболочек, к примеру, NetBeans.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Занятия по дисциплине проводятся в классе, оборудованном проектором, к каждому занятию имеются презентации, лабораторные работы проходят в компьютерном классе, оборудованном необходимым аппаратными и программными средствами. Часть материала предоставляется студенту в электронном формате. На сайте кафедры размещаются учебные пособия и презентации к лекции.