

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Белгородский Валерий Савельевич  
Должность: Ректор  
Дата подписания: 11.01.2024 12:36:52  
Уникальный программный ключ:  
8df276ee93e17c18e7bee9e7cad2d0ed9ab82473

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Российский государственный университет им. А.Н. Косыгина  
(Технологии. Дизайн. Искусство)»

Институт информационных технологий и цифровой трансформации  
Кафедра Прикладной математики и программирования

## РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

### Разработка прикладных компьютерных систем

Уровень образования	бакалавриат
Направление подготовки	01.03.02 Прикладная математика и информатика
Направленность (профиль)	Системное программирование и компьютерные технологии
Срок освоения образовательной программы по очной форме обучения	4 года
Форма обучения	очная

Рабочая программа учебной дисциплины «Разработка прикладных компьютерных систем» основной профессиональной образовательной программы высшего образования, рассмотрена и одобрена на заседании кафедры, протокол № 6 от 14.02.2023 г.

Разработчики рабочей программы учебной дисциплины:

Доцент И. В. Сеницын  
Заведующий кафедрой: О. П. Новиков

## 1. ОБЩИЕ СВЕДЕНИЯ

Учебная дисциплина «Разработка прикладных компьютерных систем» изучается в шестом семестре.

### 1.1. Форма промежуточной аттестации:

Шестой семестр - Экзамен

### 1.2. Место учебной дисциплины в структуре ОПОП

Учебная дисциплина «Разработка прикладных компьютерных систем» относится к части, формируемой участниками образовательных отношений.

Основой для освоения дисциплины являются результаты обучения по предшествующим дисциплинам:

- ИКТ в профессиональной деятельности;
- Основные методы и инструменты проектирования и разработки ПО;
- Сервисные системы обработки данных.

Результаты обучения по учебной дисциплине, используются при изучении следующих дисциплин:

- Теория алгоритмов, языков программирования и трансляторов;
- Информационная безопасность.

Результаты освоения учебной дисциплины в дальнейшем будут использованы при прохождении производственной практики и (или) выполнении выпускной квалификационной работы.

## 2. ЦЕЛИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Целями изучения дисциплины «Разработка прикладных компьютерных систем» являются:

- изучение методов и средств разработки программного обеспечения;
- освоение математически сложных алгоритмов;
- формирование навыков программирования и разработки программного обеспечения;
- обучение методам отладки и оценки качества программного кода;
- формирование у обучающихся компетенции, установленной образовательной программой в соответствии с ФГОС ВО по данной дисциплине.

Результатом обучения по учебной дисциплине является овладение обучающимися знаниями, умениями, навыками и опытом деятельности, характеризующими процесс формирования компетенции и обеспечивающими достижение планируемых результатов освоения учебной дисциплины.

### 2.1. Формируемые компетенции, индикаторы достижения компетенций, соотнесённые с планируемыми результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
ПК-1 Способен формировать новые	ИД-ПК-1.3 Использование современных информационных технологий и методов программирования	– Понимает и использует концепции процедурного программирования;

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
направления научных исследований и опытно-конструкторских разработок	при реализации научных исследований и опытно-конструкторских работ	– Понимает и использует концепции объектно-ориентированного программирования;
ПК-2 Способен разрабатывать и адаптировать прикладное программное обеспечение	ИД-ПК-2.1 Анализ инструментальных средств и методов разработки программного обеспечения и их использование для решения профессиональных задач	– Применяет современные средства и методы разработки программного обеспечения; – Использует стандартную библиотеку шаблонов при разработке программного обеспечения.
	ИД-ПК-2.2 Использование методов отладки, оценки качества, проверки работоспособности и модификации программного обеспечения	– Использует базовые методы отладки программного кода;
	ИД-ПК-2.3 Применение и реализация математически сложных алгоритмов в современных программных комплексах	– Реализует базовые алгоритмы при разработке программного обеспечения.
ПК-3 Способен участвовать в создании, модификации и сопровождении информационных систем	ИД-ПК-3.3 Использование основных алгоритмических и программных решений, связанных с объектами и системами информационных технологий	– Применяет фундаментальные знания математики при разработке алгоритмов.

### 3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Общая трудоёмкость учебной дисциплины по учебному плану составляет:

по очной форме обучения –	6	з.е.	108	час.
---------------------------	---	------	-----	------

#### 3.1. Структура учебной дисциплины для обучающихся по видам занятий

Структура и объем дисциплины									
Объем дисциплины по семестрам	форма промежуточной аттестации	всего, час	Контактная аудиторная работа, час				Самостоятельная работа обучающегося, час		
			лекции, час	практические занятия, час	лабораторные занятия, час	практическая подготовка, час	курсовая работа	самостоятельная работа обучающегося, час	промежуточная аттестация, час
6 семестр	экзамен	216	34		68			78	36
Всего:		216	34		68			78	36

## 3.2. Структура учебной дисциплины для обучающихся по разделам и темам дисциплины:

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы, час	Практическая подготовка, час		
<b>Шестой семестр</b>							
ПК-1: ИД-ПК-1.3	<b>Раздел I. Введение в JAVA</b>	x	x	x	x	16	<b>Текущий контроль успеваемости:</b> <i>Лабораторные работы (ЛБ).</i>
	Тема 1.1 Язык и платформа Java.	4		4		x	
ПК-2: ИД-ПК-2.1 ИД-ПК-2.2 ИД-ПК-2.3	Тема 1.2 Функции и технологии среды разработки.	4		4		x	
	<b>Раздел II. Основы программирования на JAVA</b>					36	
	Тема 2.1 Типы данных и переменные.	6		8		x	
ПК-3: ИД-ПК-3.3	Тема 2.2 Классы. Объектно-ориентированное программирование.	4		12		x	
	Тема 2.3 Коллекции.	4		12		x	
	<b>Раздел III. Продвинутое технологии на Java</b>	x	x	X	x	26	
	Тема 3.1 Многопоточное программирование.	6		16		x	
	Тема 3.2 Графика на JAVA	6		12		x	
	Экзамен	x	x	x	x	36	Экзамен по билетам
	<b>ИТОГО за шестой семестр</b>	<b>34</b>		<b>68</b>		<b>114</b>	

## 3.3. Краткое содержание учебной дисциплины

№ пп	Наименование раздела и темы дисциплины	Содержание раздела (темы)
<b>Раздел I</b>	<b>Раздел I. Введение в JAVA</b>	
Тема 1.1	Язык и платформа Java.	<ol style="list-style-type: none"> <li>1. Язык и платформа</li> <li>2. Расширения языка Java</li> <li>3. Строки в конструкции switch</li> <li>4. Усовершенствованный синтаксис для числовых литералов</li> <li>5. Обработка исключений</li> <li>6. Использование ресурсов в блоке try (try-with-resources)</li> <li>7. Ромбовидный синтаксис</li> <li>8. Упрощенный вызов методов с переменным количеством аргументов.</li> </ol>
Тема 1.2	Функции и технологии среды разработки.	<ol style="list-style-type: none"> <li>1. Подсказки NetBeans для эффективной разработки</li> <li>2. Автозавершение кода</li> <li>3. Шаблоны кода</li> <li>4. Визуальные индикаторы NetBeans</li> <li>5. Функция ускорения разработки HTML5</li> <li>6. Установка и первый запуск NetBeans</li> <li>7. Настройка NetBeans для разработки Java EE-приложений</li> <li>8. Развертывание первого приложения</li> <li>9. Интегрирование NetBeans со сторонним сервером приложений</li> <li>10. Интегрирование NetBeans с СУРБД стороннего производителя</li> </ol>
<b>Раздел II</b>	<b>Раздел II. Основы программирования на JAVA</b>	
Тема 2.1	Типы данных и переменные.	<ol style="list-style-type: none"> <li>1. Типы данных и переменные</li> <li>2. Операции языка Java</li> <li>3. Массивы</li> <li>4. Условные конструкции</li> <li>5. Циклы</li> <li>6. Методы</li> <li>7. Преобразования базовых типов данных</li> <li>8. Управление выполнением программы</li> <li>9. Обработка исключений</li> <li>10. Программирование графических пользовательских интерфейсов</li> <li>11. Консольный ввод/вывод в Java</li> <li>12. Рекурсивные функции</li> </ol>

Тема 2.2	Классы. Объектно-ориентированное программирование.	<ol style="list-style-type: none"> <li>1. Классы и объекты</li> <li>2. Пакеты</li> <li>3. Модификаторы доступа и инкапсуляция</li> <li>4. Статические члены и модификатор static</li> <li>5. Объекты как параметры методов</li> <li>6. Абстрактные классы</li> <li>7. Внутренние классы</li> <li>8. Интерфейсы</li> <li>9. Перечисления enum</li> <li>10. Изучение полиморфизма</li> <li>11. Приведение типов</li> <li>12. Анонимные и вложенные классы</li> <li>13. Работа с модификатором static</li> <li>14. Наследование, полиморфизм и ключевое слово super</li> <li>15. Интерфейсы в механизме обратного вызова</li> <li>16. Иерархия наследования и преобразование типов</li> <li>17. Класс Object и его методы</li> <li>18. Обобщенные типы и методы</li> </ol>
Тема 2.3	Коллекции.	<ol style="list-style-type: none"> <li>1. Коллекции</li> <li>2. Класс ArrayList и</li> <li>3. Класс LinkedList</li> <li>4. Класс HashSet</li> <li>5. Класс TreeSet</li> <li>6. Класс ArrayDeque</li> <li>7. Класс HashMap</li> <li>8. Класс TreeMap</li> <li>9. Интерфейс List</li> <li>10. Интерфейсы</li> <li>11. Comparable и Comparator</li> <li>12. Очереди</li> <li>13. Отображения</li> <li>14. Итераторы</li> <li>15. Сортировка</li> </ol>
<b>Раздел III</b>	<b>Раздел III. Продвинутое технологии на Java</b>	
Тема 3.1	Многопоточное программирование.	<ol style="list-style-type: none"> <li>1. Класс Thread</li> <li>2. Взаимодействие потоков.</li> <li>3. Методы wait и notify</li> <li>4. Семафоры</li> <li>5. Классы Exchanger и Phaser</li> <li>6. Создание и завершение потоков</li> <li>7. Синхронизация потоков. Оператор synchronized</li> <li>8. Задача "Producer-Consumer"</li> <li>9. Блокировки. Условия в блокировках</li> <li>10. Обмен между потоками</li> </ol>
Тема 3.2	Графика на JAVA	<ol style="list-style-type: none"> <li>1. Графический контекст компонентов Graphics</li> <li>2. Начертить прямую линию</li> <li>3. Задание цвета</li> <li>4. Задание цвета фона</li> <li>5. Задание прямоугольника</li> <li>6. Залить прямоугольник цветом</li> <li>7. Задание прямоугольника с закругленными углами</li> <li>8. Сформировать овал</li> <li>9. Сформировать окружность</li> <li>10. Нарисовать дугу</li> <li>11. Нарисовать многоугольник</li> <li>12. Создать объект точка</li> </ol>

		13. Определение принадлежности точки многоугольнику 14. Вывод строки 15. Задание шрифта 16. Задание цвета текста 17. Формирование графика 18. Java 2D
--	--	--

### 3.4. Организация самостоятельной работы обучающихся

Самостоятельная работа студента – обязательная часть образовательного процесса, направленная на развитие готовности к профессиональному и личностному самообразованию, на проектирование дальнейшего образовательного маршрута и профессиональной карьеры.

Самостоятельная работа обучающихся по дисциплине организована как совокупность аудиторных и внеаудиторных занятий и работ, обеспечивающих успешное освоение дисциплины.

Аудиторная самостоятельная работа обучающихся по дисциплине выполняется на учебных занятиях под руководством преподавателя и по его заданию. Аудиторная самостоятельная работа обучающихся входит в общий объем времени, отведенного учебным планом на аудиторную работу, и регламентируется расписанием учебных занятий.

Внеаудиторная самостоятельная работа обучающихся – планируемая учебная, научно-исследовательская, практическая работа обучающихся, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия, расписанием учебных занятий не регламентируется.

Внеаудиторная самостоятельная работа обучающихся включает в себя:

- изучение теоретического и практического материала по рекомендованным источникам;
- выполнение лабораторных работ;
- подготовка к промежуточной аттестации в течение семестра.

Самостоятельная работа обучающихся с участием преподавателя предусматривает проведение консультаций перед зачетом.

### 3.5. Применение электронного обучения, дистанционных образовательных технологий

При реализации программы учебной дисциплины электронное обучение и дистанционные образовательные технологии не применяются.

#### 4. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, КРИТЕРИИ ОЦЕНКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ, СИСТЕМА И ШКАЛА ОЦЕНИВАНИЯ

##### 4.1. Соотнесение планируемых результатов обучения с уровнями сформированности компетенции.

Уровни сформированности компетенции	Итоговое количество баллов в 100-балльной системе по результатам текущей и промежуточной аттестации	Оценка в пятибалльной системе по результатам текущей и промежуточной аттестации	Показатели уровня сформированности профессиональной компетенции
			ПК-1: ИД-ПК-1.3  ПК-2: ИД-ПК-2.1 ИД-ПК-2.2 ИД-ПК-2.3  ПК-3: ИД-ПК-3.3
высокий		отлично	Обучающийся: <ul style="list-style-type: none"> <li>– исчерпывающе и логически стройно излагает учебный материал, умеет связывать теорию с практикой, справляется с решением задач профессиональной направленности высокого уровня сложности, правильно обосновывает принятые решения;</li> <li>– в полной мере применяет изученные методы и средства разработки программного обеспечения при решении поставленных задач;</li> <li>– показывает высокий уровень понимания математически сложных алгоритмов и способен применять их в решении поставленных задач;</li> <li>– свободно ориентируется в программном коде, оценивает на высоком уровне качество кода, осуществляет его отладку и при возникновении необходимости, модифицирует программный код;</li> <li>– демонстрирует высокий уровень программирования и разработки программного обеспечения;</li> <li>– свободно ориентируется в учебной и профессиональной литературе;</li> <li>– дает развернутые, исчерпывающие, профессионально грамотные ответы на вопросы, в том числе, дополнительные.</li> </ul>
повышенный		хорошо	Обучающийся: <ul style="list-style-type: none"> <li>– достаточно подробно, грамотно и по существу излагает изученный материал, приводит и раскрывает в тезисной форме основные понятия;</li> </ul>



			<ul style="list-style-type: none"> <li>– использует базовые методы и средства разработки программного обеспечения при решении поставленных задач;</li> <li>– показывает базовый уровень понимания математически сложных алгоритмов и способен применять их в решении поставленных задач;</li> <li>– ориентируется в программном коде, оценивает уровень качество кода невысокой сложности, осуществляет его отладку с небольшими пробелами;</li> <li>– демонстрирует достаточно хороший уровень программирования и разработки программного обеспечения;</li> <li>– достаточно хорошо ориентируется в учебной и профессиональной литературе;</li> <li>– дает ответы на поставленные вопросы, отражающие знания теоретического материала, при этом, не допуская существенных неточностей.</li> </ul>
базовый		удовлетворительно	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– демонстрирует теоретические знания основного учебного материала дисциплины в объеме, необходимом для дальнейшего освоения образовательной программы;</li> <li>– с неточностями применяет изученные методы и средства разработки программного обеспечения при решении поставленных задач;</li> <li>– демонстрирует достаточный уровень написания программного кода;</li> <li>– демонстрирует фрагментарные знания основной учебной литературы по дисциплине;</li> <li>– дает ответы, отражающие знания на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профилю обучения.</li> </ul>
низкий		неудовлетворительно	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– демонстрирует фрагментарные знания теоретического и практического материала, допускает грубые ошибки при его изложении на занятиях и в ходе промежуточной аттестации;</li> <li>– испытывает серьезные затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, не владеет необходимыми для этого навыками и приёмами;</li> <li>– не способен применять изученные методы и средства разработки программного обеспечения при решении поставленных задач;</li> <li>– не способен программировать и разрабатывать программное обеспечение на достаточном уровне;</li> <li>– дает ответы, отражающие отсутствие знаний на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы.</li> </ul>

## 5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ, ВКЛЮЧАЯ САМОСТОЯТЕЛЬНУЮ РАБОТУ ОБУЧАЮЩИХСЯ

При проведении контроля самостоятельной работы обучающихся, текущего контроля и промежуточной аттестации по учебной дисциплине «Языки и методы программирования» проверяется уровень сформированности у обучающихся компетенций и запланированных результатов обучения по дисциплине, указанных в разделе 2 настоящей программы.

### 5.1. Формы текущего контроля успеваемости, примеры типовых заданий:

№ пп	Формы текущего контроля	Примеры типовых заданий
Раздел I, II, III	Лабораторные работы (решение задач /выполнение заданий по программированию)	<p>1) Написать программу, которая по номеру дня недели печатает его название.</p> <p>2) Каждая квартира принадлежит категории А, если никто не прописан и нет счетчика на воду; категории В, если счетчика нет, но кто-то прописан, и С, если есть счетчик. Плата за воду рассчитывается по формулам: для категории А — 3 р. за квадратный метр площади, для В — 100 р. за человека и для С - 50 р. за кубометр. Написать программу, вводящую категорию квартиры и дополнительные данные, и рассчитывающую плату за воду</p> <p>3) Написать программу, вводящую с клавиатуры целые числа и вычисляющую их произведение до тех пор, пока не будет введено число 1. В этом случае программа должна напечатать результат и завершиться.</p> <p>4) Написать программу, вводящую с клавиатуры вещественные числа до тех пор, пока не будет введено отрицательное число. В этом случае программа должна напечатать число введенных чисел (включая последнее отрицательное) и завершиться. Подсказка: нужно завести целую переменную с начальным значением 1 и в цикле каждый раз увеличивать ее на 1.</p> <p>5) Написать программу, вводящую с клавиатуры целые числа и вычисляющую их сумму до тех пор, пока эта сумма не станет больше 1000. В этом случае программа должна напечатать вычисленную сумму и завершиться.</p> <p>6) Написать программу, которая печатает таблицу функции тангенс для значений параметра от -0.5 до 0.5 с шагом 0.2.</p> <p>7) Написать программу, которая вводит с клавиатуры 10 целых чисел и печатает их сумму.</p> <p>8) Написать программу, которая вводит с клавиатуры 10 целых чисел и печатает минимальное из них.</p> <p>9) Написать программу, которая вводит с клавиатуры целое число n, затем n вещественных чисел и вычисляет количество отрицательных среди введенных чисел.</p> <p>10) Создайте приложение, в котором имеются три класса: fclass1,</p>

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p>fclass2, fclass3. В классе fclass1 содержится метод main(). Класс fclass3 наследуется от fclass1, а fclass2 – от fclass3.</p> <p>11) Создайте приложение, в котором имеются два класса: fclass1, fclass2. В классе fclass1 содержится метод main(). Создайте методы в fclass2 для доступа к закрытым переменным класса fclass1.</p> <p>12) Написать иерархию классов, описывающих имущество налогоплательщиков. Она должна состоять из абстрактного базового класса Property и производных от него классов Apartment, Car и CountryHouse. Базовый класс должен иметь поле worth (стоимость), конструктор с одним параметром, заполняющий это поле, и чисто виртуальный метод расчета налога, переопределенный в каждом из производных классов. Налог на квартиру вычисляется как 1/1000 ее стоимости, на машину – 1/200, на дачу – 1/500. Также, каждый производный класс должен иметь конструктор с одним параметром, передающий свой параметр конструктору базового класса. В методе main завести массив из 7 указателей на Property и заполнить его объектами производных классов (первые 3 – Apartment, следующие 2 – Car и последние 2 – CountryHouse). Вывести на экран величину налога для всех 7 объектов.</p> <p>13) Напишите программу, создающую две метки в окне. Как они будут располагаться относительно окна и друготносительно друга? А если менять размер окна?</p> <p>14) Написать программу, при запуске которой в главном окне появляется кнопка, на которой написано 10, и при каждом нажатии это число уменьшается на 1, до тех пор, пока оно не станет равно 1. В этом случае при нажатии на кнопку программа завершается.</p> <p>15) Написать программу, при запуске которой в главном окне появляется две кнопки, каждая из которых включает/отключает другую (для включения кнопки нужно вызвать метод setEnabled(true), для отключения - setEnabled(false)).</p> <p>16) Создайте форму размером 500x500 и нарисуйте на ней какую-нибудь картинку.</p> <p>17) Сумматор. Создайте приложение, которое представляет собой форму 200x150. На форме разместите три текстовых поля и одну кнопку с надписью «Расчет». При нажатии на кнопку значения, введенные в первые два текстовых поля, складываются и результат записывается в третье поле.</p>

## 5.2. Критерии, шкалы оценивания текущего контроля успеваемости:

Наименование оценочного средства (контрольно-оценочного мероприятия)	Критерии оценивания	Шкалы оценивания	
		Зачет/Незачет	Пятибалльная система
Лабораторные работы (решение задач /выполнение заданий по программированию)	Обучающийся демонстрирует грамотное решение всех задач по программированию (программы работают и не выдают ошибок), использование правильных методов решения при незначительных погрешностях, не влияющих на работоспособность написанного кода.	зачет	5
	Продемонстрировано использование правильных методов написания программ при наличии 1-2 существенных ошибок.	зачет	4
	Обучающийся использует верные методы решения при написании программ, но выполненные верно задания в большинстве случаев отсутствуют.	зачет	3
	Обучающийся использует неверные методы решения при написании программ, большинство задач выполнены некорректно.	незачет	2

## 5.3. Промежуточная аттестация:

Форма промежуточной аттестации	Типовые контрольные задания и иные материалы для проведения промежуточной аттестации:
Экзамен	<p>В билете 2 теоретических вопроса и задача.</p> <p>Перечень вопросов к экзамену по дисциплине “Разработка прикладных компьютерных систем”</p> <ol style="list-style-type: none"> <li>1. Программирование на JAVA. Основные достоинства и недостатки языка</li> <li>2. Основные особенности языка JAVA</li> <li>3. Алгоритм работы системы программирования на JAVA</li> <li>4. Жизненным цикл программы на JAVA</li> <li>5. Понятие среды выполнения Java. Версии языка программирования JAVA.</li> <li>6. Спецификация языка программирования JAVA, стандартные библиотеки.</li> <li>7. Типы JAVA -приложений</li> <li>8. Методы объектно-ориентированного программирования применяемого JAVA</li> <li>9. Java переменные</li> <li>10. Правила именования переменных в JAVA</li> </ol>

	<ol style="list-style-type: none"><li>11. Типы данных в JAVA</li><li>12. Простейшая программа на Java</li><li>13. Лексические основы</li><li>14. Комментарии</li><li>15. Лексемы</li><li>16. Идентификаторы</li><li>17. Ключевые слова</li><li>18. Литералы (константы)</li><li>19. Java класс Math</li><li>20. Псевдослучайные числа</li><li>21. Ввод информации с клавиатуры</li><li>22. Условные операторы</li><li>23. Оператор выбора</li><li>24. Цикл типа «n раз» ( For )</li><li>25. Цикл типа «пока». (While)</li><li>26. Массивы в JAVA</li><li>27. Класс java.util.Arays</li><li>28. Сортировка массива</li><li>29. Копирование массивов</li><li>30. Вывод одномерных массивов с помощью метода toString()</li><li>31. Вывод многомерных массивов с помощью метода deepToString()</li><li>32. Заполнение массива</li><li>33. Сравнение массивов</li><li>34. Перебор значений массива</li><li>35. Сложить два массива</li><li>36. Взять часть массива</li><li>37. Работа со строками (классы String и StringBuffer)</li><li>38. Создание и инициализация объекта класса StringBuffer</li><li>39. Сравнение строк</li><li>40. Поиск в строках</li><li>41. Извлечение символов и подстрок из строки</li><li>42. Модификация строк</li><li>43. Кодировка</li><li>44. Анализ программы. Пробелы</li><li>45. Анализ программы. Комментарии</li></ol>
--	--

	<ul style="list-style-type: none"><li>46. Анализ программы. Лексемы</li><li>47. Анализ программы. Идентификаторы</li><li>48. Анализ программы. Ключевые слова. Разделители.</li><li>49. Операторы</li><li>50. Операторы присваивания и сравнения</li><li>51. Арифметические операции</li><li>52. Логические операторы</li><li>53. Объектно-ориентированное программирование на JAVE</li><li>54. Графический контекст компонентов</li><li>55. Начертить прямую линию</li><li>56. Задать цвет</li><li>57. Задать rgb цвета</li><li>58. Задать цвет фона</li><li>59. Залить прямоугольник цветом. Нарисовать прямоугольник с закругленными углами</li><li>60. Нарисовать овал. Нарисовать окружность</li><li>61. Нарисовать дугу</li><li>62. Нарисовать многоугольник</li><li>63. Создать объект точки</li><li>64. Вывести строку. Задать шрифт</li><li>65. Задать цвет текста. Начертить график функции.</li><li>66. Подключение базы данных MySQL с помощью JDBC</li><li>67. Напишем программу на Java, которая использует базу данных</li><li>68. Многопоточность на JAVA. Процессы</li><li>69. Многопоточность на JAVA. Потoki</li><li>70. Запуск потоков</li><li>71. Завершение процесса и демоны</li><li>72. Завершение потоков</li><li>73. Interruption</li><li>74. Метод Thread.sleep(). Метод yield().Метод join().</li><li>75. Приоритеты потоков</li><li>76. Некоторые полезные методы класса Thread.</li></ul>
--	---

## 5.4. Критерии, шкалы оценивания промежуточной аттестации учебной дисциплины/модуля:

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
		Зачет/незачет	Пятибалльная система
Экзамен	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– демонстрирует знания, отличающиеся глубиной и содержательностью, дает полный исчерпывающий ответ, как на основные вопросы билета, так и на дополнительные;</li> <li>– свободно выполняет практические задания повышенной сложности, предусмотренные дисциплиной. Написанный программный код по выполнению практического задания не содержит ошибок и работает корректно.</li> </ul>		5
	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– показывает достаточное знание учебного материала, но допускает несущественные фактические ошибки, которые способен исправить самостоятельно, благодаря наводящему вопросу;</li> <li>– успешно выполняет предусмотренные в дисциплине практические задания средней сложности, написанный программный код по выполнению практического задания работает корректно, допущены лишь несущественные ошибки, которые исправимы в процессе обсуждения выполненного задания.</li> </ul>		4
	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– показывает знания фрагментарного характера, которые отличаются поверхностностью и малой содержательностью, допускает фактические грубые ошибки;</li> <li>– справляется с выполнением практических заданий, предусмотренных дисциплиной, знаком с основной литературой, рекомендованной программой, допускает погрешности и ошибки при теоретических ответах и в ходе практической работы при написании программ.</li> </ul>		3
	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– демонстрирует существенные пробелы в знаниях основного учебного материала, допускает принципиальные ошибки в выполнении предусмотренных программой практических заданий.</li> <li>– на большую часть дополнительных вопросов по содержанию экзамена затрудняется дать ответ или не дает верных ответов.</li> </ul>		2

### 5.5. Система оценивания результатов текущего контроля и промежуточной аттестации.

Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.

Форма контроля	Зачет/Незачет	Пятибалльная система
<b>Шестой семестр</b>		
Текущий контроль:	<i>x</i>	<i>x</i>
Лабораторные работы (решение задач/выполнение заданий по программированию) по разделу I,II,III	<i>Зачет/Незачет</i>	2 – 5
Промежуточная аттестация (Экзамен)		отлично хорошо удовлетворительно неудовлетворительно
<b>Итого за семестр (дисциплину)</b> Экзамен		

## 6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Реализация программы предусматривает использование в процессе обучения следующих образовательных технологий:

- поиск и обработка информации с использованием сети Интернет;
- дистанционные образовательные технологии в случае производственной необходимости;
- применение электронного обучения в случае производственной необходимости.

## 7. ПРАКТИЧЕСКАЯ ПОДГОТОВКА

Практическая подготовка в рамках учебной дисциплины реализуется при проведении практических занятий.

## 8. ОРГАНИЗАЦИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

При обучении лиц с ограниченными возможностями здоровья и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуального обучения, применение соответствующих методик по работе с инвалидами, использование средств дистанционного общения, проведение дополнительных индивидуальных консультаций по изучаемым теоретическим вопросам и практическим занятиям, оказание помощи при подготовке к промежуточной аттестации.

При необходимости рабочая программа дисциплины может быть адаптирована для обеспечения образовательного процесса лицам с ограниченными возможностями здоровья, в том числе для дистанционного обучения.

Учебные и контрольно-измерительные материалы представляются в формах, доступных для изучения студентами с особыми образовательными потребностями с учетом нозологических групп инвалидов.

Для подготовки к ответу на практическом занятии, студентам с ограниченными возможностями здоровья среднее время увеличивается по сравнению со средним временем подготовки обычного студента.

Для студентов с инвалидностью или с ограниченными возможностями здоровья форма проведения текущей и промежуточной аттестации устанавливается с учетом индивидуальных



психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Промежуточная аттестация по дисциплине может проводиться в несколько этапов в форме рубежного контроля по завершению изучения отдельных тем дисциплины. При необходимости студенту предоставляется дополнительное время для подготовки ответа на зачете или экзамене.

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации обучающихся создаются, при необходимости, фонды оценочных средств, адаптированные для лиц с ограниченными возможностями здоровья и позволяющие оценить достижение ими запланированных в основной образовательной программе результатов обучения и уровень сформированности всех компетенций, заявленных в образовательной программе.

## 9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Материально-техническое обеспечение дисциплины при обучении с использованием традиционных технологий обучения.

Наименование учебных аудиторий, лабораторий, мастерских, библиотек, спортзалов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.	Оснащенность учебных аудиторий, лабораторий, мастерских, библиотек, спортивных залов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.
<b>119071, г. Москва, Малый Калужский переулок, дом 1, строение 2</b>	
аудитории для проведения занятий лекционного типа	комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – ноутбук; – проектор; – проекционный экран.
аудитории для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – ноутбук; – проектор; – проекционный экран; – персональные компьютеры для обучающихся.
<b>Помещения для самостоятельной работы обучающихся</b>	<b>Оснащенность помещений для самостоятельной работы обучающихся</b>
читальный зал библиотеки	– компьютерная техника; подключение к сети Интернет.

## 10. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

№ п/п	Автор(ы)	Наименование издания	Вид издания (учебник, УП, МП и др.)	Издательство	Год издания	Адрес сайта ЭБС или электронного ресурса	Количество экземпляров в библиотеке Университета
10.1 Основная литература, в том числе электронные издания							
1	Гуськова, О.И.	Объектно ориентированное программирование в Java	Учебное пособие	М. : МПГУ	2019	<a href="http://znanium.com/catalog.php?bookinfo=1020593">http://znanium.com/catalog.php?bookinfo=1020593</a>	
2	Соколова В.В.	Разработка мобильных приложений	Учебное пособие	Томск: Изд-во Томского политех. университета	2020	<a href="http://znanium.com/catalog.php?bookinfo=701720">http://znanium.com/catalog.php?bookinfo=701720</a>	
3	Кузьмина Т.М.	Объектно-ориентированное программирование. Конспект лекций	Учебное пособие	М.:МГУДТ	2015	<a href="http://znanium.com/catalog.php?bookinfo=791800">http://znanium.com/catalog.php?bookinfo=791800</a>	
4	Кувшинов Д. Р.	Основы программирования	Учебное пособие	М.: Юрайт	2021	<a href="https://urait.ru/bcode/473570">https://urait.ru/bcode/473570</a>	-
10.2 Дополнительная литература, в том числе электронные издания							
1	Зыков С. В.	Программирование. Объектно-ориентированный подход	Учебник и практикум	М.: Юрайт	2021	<a href="https://urait.ru/bcode/470281">https://urait.ru/bcode/470281</a>	-
2	Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю.	Технологии и методы программирования	Учебное пособие	М.: Юрайт	2021	<a href="https://urait.ru/bcode/469759">https://urait.ru/bcode/469759</a>	-
3	Кувшинов Д. Р.	Основы программирования	Учебное пособие	М.: Юрайт	2021	<a href="https://urait.ru/bcode/473570">https://urait.ru/bcode/473570</a>	-

## 11. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

11.1. Ресурсы электронной библиотеки, информационно-справочные системы и профессиональные базы данных:

№ пп	Электронные учебные издания, электронные образовательные ресурсы
1.	ЭБС «Лань» <a href="http://www.e.lanbook.com/">http://www.e.lanbook.com/</a>
2.	«Znanium.com» научно-издательского центра «Инфра-М» <a href="http://znanium.com/">http://znanium.com/</a>
3.	Электронные издания «РГУ им. А.Н. Косыгина» на платформе ЭБС «Znanium.com» <a href="http://znanium.com/">http://znanium.com/</a>
4.	Образовательная платформа «Юрайт» <a href="https://urait.ru/">https://urait.ru/</a>
5.	Электронные ресурсы «Polpred.com Обзор СМИ» <a href="https://www.polpred.com/">https://www.polpred.com/</a>
6.	Электронные ресурсы «Национальной электронной библиотеки» («НЭБ») <a href="https://rusneb.ru/">https://rusneb.ru/</a>
Профессиональные базы данных, информационные справочные системы	
1.	Информационно-аналитическая система SCIENCE INDEX (включенная в научный информационный ресурс eLIBRARY.RU) <a href="https://www.elibrary.ru/">https://www.elibrary.ru/</a>
2.	База данных Springer eBooks Collections издательства Springer Nature. Платформа Springer Link: <a href="https://rd.springer.com/">https://rd.springer.com/</a>
3.	Электронный ресурс Freedom Collection издательства Elsevier <a href="https://sciencedirect.com/">https://sciencedirect.com/</a>
4.	База данных научного цитирования Scopus издательства Elsevier <a href="https://www.scopus.com/">https://www.scopus.com/</a>
5.	База данных ORBIT IPBI (Platinum Edition) компании Questel SAS <a href="https://www.orbit.com/">https://www.orbit.com/</a>
6.	База данных Web of Science компании Clarivate Analytics <a href="https://www.webofscience.com/wos/woscc/basic-search">https://www.webofscience.com/wos/woscc/basic-search</a>
7.	База данных CSD-Enterprise компании The Cambridge Crystallographic Data Center <a href="https://www.ccdc.cam.ac.uk/">https://www.ccdc.cam.ac.uk/</a>
8.	Научная электронная библиотека «elibrary.ru» <a href="https://www.elibrary.ru/">https://www.elibrary.ru/</a>
9.	База данных издательства SpringerNature <a href="https://link.springer.com/">https://link.springer.com/</a> <a href="https://www.springerprotocols.com/">https://www.springerprotocols.com/</a> <a href="https://materials.springer.com/">https://materials.springer.com/</a> <a href="https://link.springer.com/search?facet-content-type=%ReferenceWork%22">https://link.springer.com/search?facet-content-type=%ReferenceWork%22</a> <a href="http://zbmath.org/">http://zbmath.org/</a> <a href="http://npg.com/">http://npg.com/</a>

11.2. Перечень программного обеспечения

№п/п	Программное обеспечение	Реквизиты подтверждающего документа/ Свободно распространяемое
1.	Windows 10 Pro, MS Office 2019	контракт № 18-ЭА-44-19 от 20.05.2019
2.	Microsoft Visual StudioЮ	контракт № 18-ЭА-44-19 от 20.05.2019
3.	PHP, MYSQL, WEB APACHE	

**ЛИСТ УЧЕТА ОБНОВЛЕНИЙ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

В рабочую программу учебной дисциплины внесены изменения/обновления и утверждены на заседании кафедры:

<b>№ пп</b>	<b>год обновления РПД</b>	<b>характер изменений/обновлений с указанием раздела</b>	<b>номер протокола и дата заседания кафедры</b>