

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тольяттинский государственный университет»

Б1.В.ДВ.03.01  
(индекс дисциплины)

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**Разработка приложений на платформе Java (Джава) 1**

(наименование дисциплины)

по направлению подготовки

01.03.02 Прикладная математика и информатика

направленность (профиль)

Компьютерные технологии и математическое моделирование

Форма обучения: очная

Год набора: 2026

Общая трудоемкость: 6 ЗЕ

**Распределение часов дисциплины по семестрам**

Семестр	6	Итого
Форма контроля	экзамен	
Вид занятий		
Лекции	32	32
Лабораторные		
Практические	48	48
Руководство: курсовые работы (проекты) / РГР		
Промежуточная аттестация	0,35	0,35
Контактная работа	80.35	80.35
Самостоятельная работа	100	100
Контроль	35.65	35.65
<b>Итого</b>	<b>216</b>	<b>216</b>

Рабочую программу составил(и):

Доцент института цифровых технологий, к.п.н., Ерофеева Е.А.

---

*(должность, ученое звание, степень, Фамилия И.О.)*

---

*(должность, ученое звание, степень, Фамилия И.О.)*

Рецензирование рабочей программы дисциплины:

☐

Отсутствует

☐

Рецензент

---

*(должность, ученое звание, степень, Фамилия И.О.)*

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана  
направления подготовки

01.03.02 Прикладная математика и информатика

---

**Срок действия рабочей программы дисциплины до «31» августа 2030 г.**

УТВЕРЖДЕНО

на заседании Института цифровых технологий

---

**(протокол заседания № 1 от «05» сентября 2025 г.).**

## 1. Цель освоения дисциплины

Цель освоения дисциплины –освоение технологии разработки, тестирования и развертывания программного обеспечения на платформе Java SE

## 2. Место дисциплины в структуре ОПОП ВО

Дисциплины, учебные курсы, на освоении которых базируется данная дисциплина: Программирование на языках высокого уровня; Алгоритмы и структуры данных; Архитектура компьютеров и операционные системы; Объектно-ориентированное программирование.

Дисциплины, учебные курсы, для которых необходимы знания, умения, навыки, приобретаемые в результате изучения данной дисциплины: Разработка приложений на платформе Java 2; производственная практика; подготовка выпускной квалификационной работы.

## 3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
ПК-5 Способен осуществлять выбор языка программирования и моделировать решение для реализации программного обеспечения	ПК-5.1 Знает принципы выбора языка программирования и моделирования решений для реализации программного обеспечения	Знать: технологии моделирования программного обеспечения Уметь: применять технологии моделирования программного обеспечения Владеть :навыками моделирования программного обеспечения
	ПК-5.2 Умеет использовать знания по выбору языка программирования и моделированию решений для реализации программного обеспечения	Знать: технологию разработки программного обеспечения на языках программирования Уметь: выбирать и моделировать решения по разработке программного обеспечения на языках программирования Владеть: навыками реализации программного обеспечения на языках программирования
	ПК-5.3 Владеет навыками выбора языка программирования и моделирования решений для реализации программного обеспечения	Знать: технологии моделирования решения для реализации программного обеспечения на языках программирования Уметь: моделировать решения для реализации программного обеспечения на языках программирования Владеть: инструментом моделирования решения для реализации программного обеспечения на языках программирования

#### 4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наимено- вание оценочного средства)
1	Лекция 1	Введение в Java.	6	2		-	
1	Практическое занятие 1	Утилиты JDK	6	2	6	-	Отчет по практической работе
1	Лекция 2	Базовые конструкции Java	6	2		-	
1	Практическое занятие 2	Базовые конструкции Java	6	2	8	-	Отчет по практической работе
1	Лекция 3	Алгоритмические конструкции Java	6	2		-	
1	Практическое занятие 3	Алгоритмические конструкции Java	6	2	10	-	Отчет по практической работе
1	Лекция 4	Массивы	6	2		-	
1	Практическое занятие 4	Массивы	6	2	8	-	Отчет по практической работе
2	Лекция 5	Объектно-ориентированное программи- рование в Java.	6	2		-	
2	Практическое занятие 5	Объектно-ориентированное программи- рование в Java.	6	4	8	-	Отчет по практической работе
2	Лекция 6	Пакеты и Интерфейсы	6	2		-	
2	Практическое занятие 6	Пакеты и Интерфейсы	6	2	8	-	Отчет по практической работе
2	Лекция 7	Внутренние классы	6	2		-	
2	Практическое занятие 7	Внутренние классы	6	2	8	-	Отчет по практической работе
2	Лекция 8	Generic type	6	4		-	

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наимено- вание оценочного средства)
2	Практическое занятие 8	Generic type	6	2	8		Отчет по практической работе
2	Лекция 9	Лямбда-выражения	6	4		-	
2	Практическое занятие 9	Лямбда-выражения	6	4	10	-	Отчет по практической работе
2	Лекция 10	Исключения. Отладка Java программ	6	4		-	
2	Практическое занятие 10	Исключения. Отладка Java программ	6	2	8	-	Отчет по практической работе
2	Лекция 11	Работа со строками в Java	6	4		-	
2	Практическое занятие 11	Работа со строками в Java	6	4	10	-	Отчет по практической работе
2	Лекция 12	Работа с датой и временем в Java	6	2		-	
	Практическое занятие 12	Работа с датой и временем в Java	6	4	8		Отчет по практической работе
	Сам	Самостоятельная работа	6	100			
	ПА		6	0,35			
	Контроль		6	35,65			
						-	
<b>Итого:</b>				<b>216</b>	<b>100</b>		

#### Схема расчета итогового балла: по накопительному рейтингу

Текущий рейтинг (все занятия и промежуточные тесты) + Результат итогового теста и все делится на 2

## **5. Образовательные технологии**

В рамках изучения дисциплины «Разработка приложений на платформе Java 1» предусмотрено использование следующих образовательных технологий:

- технология традиционного обучения: лекции и практические работы, самостоятельная работа;
- технология проектного обучения: реализация и защита отчетов по практическим работам.

## **6. Методические указания по освоению дисциплины**

### **6.1. Рекомендации по подготовке к лекционным занятиям**

Изучение дисциплины требует систематического и последовательного накопления знаний, следовательно, пропуски отдельных тем не позволяют глубоко освоить предмет.

В ходе лекционных занятий рекомендуется конспектировать учебный материал, обращая внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации, положительный опыт в ораторском искусстве. Желательно оставить в рабочих конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

В ходе подготовки к лекциям изучить основную литературу, ознакомиться с дополнительной литературой, публикациями в Интернет-источниках, периодических изданиях. При этом учесть рекомендации преподавателя и требования учебной программы. Дорабатывать свой конспект лекции, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной учебной программой.

Студент может дополнить список использованной литературы современными источниками, не представленными в списке рекомендованной литературы, и в дальнейшем использовать собственные подготовленные учебные материалы при написании курсовых и выпускных квалификационных работ.

### **6.2. Рекомендации по подготовке к практическим занятиям**

Студентам следует:

- до очередного практического занятия по рекомендованным литературным источникам проработать теоретический материал, соответствующей темы занятия;
- при подготовке к практическим занятиям следует обязательно использовать не только лекции, учебную литературу, но и другие источники;
- в начале занятий задать преподавателю вопросы по материалу, вызвавшему затруднения в его понимании и освоении при решении задач, заданных для самостоятельного решения;
- на занятии доводить каждую задачу до окончательного решения, демонстрировать понимание проведенных расчетов (анализов, ситуаций), в случае затруднений обращаться к преподавателю.

Для того чтобы практические занятия приносили максимальную пользу, необходимо помнить, что упражнение и решение задач проводятся по рассмотренному на лекциях материалу и связаны, как правило, с детальным разбором отдельных вопросов лекционного курса. Следует подчеркнуть, что только после усвоения лекционного материала с определенной точки зрения (а именно с той, с которой он излагается на лекциях) он будет закрепляться студентом на практических занятиях как в результате обсуждения и анализа лекционного материала, так и с помощью решения проблемных ситуаций, задач. При этих условиях студент не

только хорошо усвоит материал, но и научиться применять его на практике, а также получить дополнительный стимул (и это очень важно) для активной проработки лекции.

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

### **6.3. Рекомендации по подготовке к экзамену**

Подготовка к экзамену способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На экзамене студент демонстрирует то, что он приобрел в процессе обучения по конкретной учебной дисциплине.

На консультации перед экзаменом студенты должны быть ознакомлены с основными требованиями и получить ответы на возникающие в процессе подготовки вопросы.

Необходимо ориентировать студентов на систематическую подготовку к занятиям в течение семестра, что позволит использовать время экзаменационной сессии для систематизации знаний.

## 7. Оценочные средства

### 7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
6	ПК-5	Вопросы для собеседования по модулю 1
		Комплект отчетов по практическим работам 1-4
6	ПК-5	Вопросы для собеседования по модулю 2
		Комплект отчетов по практическим работам 5-12

### 7.2. Типовые задания или иные материалы, необходимые для текущего контроля

#### 7.2.1. Комплект отчетов по практическим работам (примеры)

(наименование оценочного средства)

##### Практическая работа 1. Утилиты Java

**Цель работы:** ознакомиться с основными утилитами Java, научиться компилировать, выполнять программы, используя утилиты `javac`, `java`, научиться создавать, модифицировать архивы Java, запускать программы из `jar`-файлов.

##### Задание:

1. Создайте текстовый файл `Test.txt` и запишите в него следующее:

```
public class Test
{
    public static void main(String[] args)
    {
        System.out.println("Hello from Test!");
    }
}
```

2. Переименуйте файл в `Test.java` и запустите его на выполнение, запишите в отчет команды и приведите скриншот выполнения

Решение: `java Test.java`

3. Скомпилируйте файл, запишите в отчет команды и приведите скриншот выполнения

Решение: `javac Test.java`

4. Перепишите код в файле таким образом, чтобы запускаемая программа выводила передаваемые в неё аргументы, запишите в отчет команды и приведите скриншот выполнения

Пример:

```
C:\Test-java>java Test arg1 arg2 123 456
arg1
arg2
123
456
```

Решение: в файле вместо `sout` пишем цикл, где обращаемся к элементам массива `args` через индекс и выводим их на консоль через `sout`

5. Создайте две копии файла `Test`. Переименуйте полученные файлы соответственно `Test1`, `Test2`, `Test3`. При запуске каждого из файлов, на консоль должно выводиться соответствующее сообщение:

Hello from Test1, Hello from Test2, ...

В текущей директории создайте каталог `bin`.



Скомпилируйте полученные файлы java таким образом, чтобы результат компиляции - файлы с расширением class, оказались в папке bin. Выполните это действие за одну команду, запишите в отчет команду и приведите скриншот выполнения

Решение: `javac -d bin *.java`

Возможные проблемы:

- Имя файла должно соответствовать имени класса
- Возможны проблемы с доступом, запустите cmd от администратора
- Не забудьте создать папку bin вручную, команда не создаёт папки автоматически

6. Удалите файлы с расширением class таким образом, чтобы они остались только в каталоге bin. Перейдите из текущей рабочей директории в любую другую внешнюю директорию. Из новой директории запустите на выполнение любой из файлов Test, запишите в отчет команды и приведите скриншот выполнения

Решение: `java -classpath Test-java\bin Test1`

7. Упакуйте полученные файлы class в архив jar таким образом, чтобы при запуске архива на консоль выводилось сообщение из программы Test2 (Hello from Test2), запишите в отчет команды и приведите скриншот выполнения

Решение: `jar cfe app.jar Test2 *.class` (находясь в каталоге bin)  
`java -jar app.jar`

8. Из этого же архива (не создавая новый) запустите на выполнение программу Test1, запишите в отчет команды и приведите скриншот выполнения

Решение: `java -cp app.jar Test1`

9. В каталоге bin создайте 3 новых каталога – t1, t2, t3. В каждый из них поместите соответственно файлы Test. Затем упакуйте каталог bin в архив jar с сохранением структуры каталогов, запишите в отчет команды и приведите скриншот выполнения.

10. Заполните отчет, включающий следующую структуру:

- Название работы
- **Цель работы**
- **Формулировка задания**
- **Исходный код с комментариями**
- **Вывод программы** (если есть консольный вывод)
- **Объяснение кода** (что делает программа, какие интерфейсы использованы и почему)
- Выводы

### Требования к оформлению

Отчет должен содержать подробное описание (включая иллюстративный материал) последовательности действий, проделанных обучающимся для выполнения заданий.

### Процедура оценивания

Оценка выполненной работы проводится по критериям:

1. Наличие всей существенной информации по работе
2. Точность и полнота предоставляемых сведений
3. Непротиворечивость приводимой информации
4. Правильность интерпретаций и выводов, которые сделаны по результатам работы
5. Степень достижения обучающимся поставленной цели
6. Обоснованность применяемого решения
7. Грамотность (содержательная) используемых формулировок

**Критерии оценки за отчеты по практическим работам:**

Формы теку-	Критерии и нормы оценки
-------------	-------------------------

щего контроля	
Отчеты по практическим работам 1 - 3	10 баллов – задание выполнено в полном объеме без замечаний 9 баллов – задание выполнено в полном объеме, присутствуют замечания 8 баллов – задание выполнено в объеме 70%, замечаний нет. 6 баллов – задание выполнено в объеме 70%, присутствуют замечания. 5 баллов – задание выполнено в объеме 50%, замечаний нет. 4 балла – задание выполнено в объеме 50%, присутствуют замечания. 3 балла - задание выполнено в объеме менее 50%, замечаний нет. 1 балл – задание выполнено в объеме менее 50%, присутствуют замечания. 0 баллов – задание не выполнено.
Отчеты по практическим работам 4 - 8	12 баллов – задание выполнено в полном объеме без замечаний 10 баллов – задание выполнено в полном объеме, присутствуют замечания 9 баллов – задание выполнено в объеме 70%, замечаний нет. 7 баллов – задание выполнено в объеме 70%, присутствуют замечания. 5 баллов – задание выполнено в объеме 50%, замечаний нет. 4 балла – задание выполнено в объеме 50%, присутствуют замечания. 3 балла - задание выполнено в объеме менее 50%, замечаний нет. 1 балл – задание выполнено в объеме менее 50%, присутствуют замечания. 0 баллов – задание не выполнено.

## 7.2.2. Комплект тестовых заданий

(наименование оценочного средства)

1. Технология Java представляет собой:
  - ☒ высокоуровневый объектно-ориентированный язык программирования
  - ☒ программную платформу
  - ☐ низкоуровневый язык программирования
  - ☐ язык сценариев
2. Что такое JVM (Java Virtual Machine)?
  - ☒ виртуальная машина Java
  - ☐ компилятор
  - ☐ набор специальных библиотек
  - ☐ техническое обеспечение
3. Исходные файлы (.java) содержат:
  - ☒ исходный код программ на Java
  - ☐ скомпилированные Java- програм - байтовые коды, определенные спецификацией Java
  - ☐ потоковое представление текущего состояния объектов
  - ☐ наборы файлов, которые могут быть представлены в упакованном виде
4. Файлы классов (.class) содержат:
  - ☒ скомпилированные Java- програм - байтовые коды, определенные спецификацией Java
  - ☐ исходный код программ на Java

- потоковое представление текущего состояния объектов
- наборы файлов, которые могут быть представлены в упакованном виде

5. Файлы сериализации (.ser) содержат:

- ⊙ потоковое представление текущего состояния
- исходный код программ на Java
- скомпилированные Java- программ - байтовые коды, определенные спецификацией Java
- наборы файлов, которые могут быть представлены в упакованном виде

6. Файлы архивов (.jar) содержат:

- потоковое представление текущего состояния
- исходный код программ на Java
- скомпилированные Java- программ - байтовые коды, определенные спецификацией Java
- ⊙ наборы файлов, которые могут быть представлены в упакованном виде

7. Java Development Kit (JDK) -это:

- виртуальная машина Java
- исходный код программ на Java
- не является компонентой Java
- ⊙ набор библиотек и утилит для разработки программ на Java

8. Назначение утилиты javac?

- утилита для запуска в режиме командной строки откомпилированных программ-приложений
- утилита для запуска на исполнение и отладку апплетов без браузера. При этом не гарантируется работоспособность отлаженного апплета в браузере
- отладчик программ, написанных на языке Java
- ⊙ компилятор в режиме командной строки для программ, написанных на языке Java

9. Что будет, если ввести команду

`jar -cfv jar.jar Hello.java Hello.class`

- запустится класс Hello из jar-архива
- файл Hello.java скомпилируется и будет создан файл Hello.class
- ⊙ будет создан jar архив с файлами Hello.java и Hello.class
- ошибка (Error)

10. Параметр -cp позволяет?

- ☒ указать путь, по которому можно найти классы для компиляции
- ☒ указать путь, по которому можно найти классы для запуска JVM
- ☐ вывести версию компилятора
- ☐ вывести версию исходного кода

11. Что произойдет при использовании данной команды

`java HelloWorld`

- компиляция jar файла «HelloWorld»
- ⊙ запустится байт-код «HelloWorld»

- компиляция исходного Java-кода «HelloWorld» в байт-код
- ошибка

12. Что выполняет данная команда `javac hellofromfirst.java`?

- Ⓐ компиляция исходного Java-кода с именем «hellofromfirst» в байт-код с таким же названием
- компиляция исходного Java-кода с именем «hellofromfirst» в jar с таким же названием
- компиляция исходного Java-кода с именем «hellofromfirst» в манифест с таким же названием
- компиляция исходного Java-кода с именем «hellofromfirst» в байт-код с другим названием

13. Базовый пакет `java.lang`?

- поддержка вычислений с целыми числами произвольной длины а так же с числами с плавающей точкой произвольной точности
- содержит важные классы для работы со структурами данных (работа с событиями и датами)
- Ⓐ содержит базовые классы языка `java`
- поддержка базовых средств ввода/вывода

14. Базовый пакет `java.io`?

- поддержка вычислений с целыми числами произвольной длины а так же с числами с плавающей точкой произвольной точности
- содержит важные классы для работы со структурами данных (работа с событиями и датами)
- содержит базовые классы языка `java`
- Ⓐ поддержка базовых средств ввода/вывода

15. Пакет (`package`) в Java это?

- Ⓐ контейнер, который используется для того, чтобы изолировать имена классов
- базовый класс языка `java`
- базовый метод языка `java`
- интерфейс

16. Выберите правильный вариант значений переменных `a, b, c, d` чтобы результат выражения `a && (b || c) ^ d` являлся ложью?

- `a=false, b=false, c=false, d=true`
- Ⓐ все ответы верны
- `a=true, b=true, c=false, d=true`
- `a=false, b=true, c=false, d=true`

17. Выберите правильный вариант значений переменных `a, b, c, d` чтобы результат выражения `a && b || c ^ d` являлся истиной?

- `a=false, b=false, c=true, d=false`
- Ⓐ все ответы верны
- `a=true, b=true, c=false, d=true`

- a=false, b=false, c=false, d=true

18. Что означает данный оператор >> ?

- оператор беззнакового правого побитового сдвига.
- оператор левого побитового сдвига
- знак больше
- ⊙ оператор правого побитового сдвига с сохранением знака отрицательного числа

19. Что означает данный оператор >>> ?

- ⊙ оператор беззнакового правого побитового сдвига.
- знак больше
- оператор левого побитового сдвига
- оператор правого побитового сдвига с сохранением знака отрицательного числа

20. Что означает данный оператор << ?

- оператор беззнакового правого побитового сдвига.
- оператор беззнакового левого побитового сдвига
- знак меньше
- ⊙ оператор левого побитового сдвига

21. Какое значение примет переменная result после выполнения программы?

```
int[] m = {4, 8, 10, 16, 12};  
int result = 0;  
for(int i=0; i<4; i++) {  
    result += m[i];  
}  
System.out.println(result);
```

- 0
- 4
- 50
- ⊙ 38

22. Укажите правильный синтаксис тернарного оператора в Java

- ⊙ логическое\_условие ? выражение1 : выражение2;
- if логическое\_условие ? выражение1 : выражение2;
- if логическое\_условие then выражение1 else выражение2;
- do ... while

23. В каких случаях применяют оператор switch?

- ⊙ к известному числу возможных ситуаций выбора
- применяют, когда число ситуаций выбора не известно
- когда происходит дублирование значений выбора
- данный оператор в программировании не используется

24. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(firstNum++);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☒ postfix: a
- ☐ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11

25. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(++firstNum);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☐ postfix: a
- ☒ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11

26. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(firstNum--);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☒ postfix: a
- ☐ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11

27. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(--firstNum);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☐ postfix: a
- ☐ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11
- ☐ postfix: 9

28. Дан код.

```
int a= 10;  
int b= 10;  
String res = Integer.toHexString(a&b);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☒ a
- ☐ 1100
- ☐ 1011
- ☐ 10

29. Дан код.

```
int a= 10;  
int b= 11;  
String res = Integer.toHexString(a|b);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☒ b
- ☐ a
- ☐ 1001
- ☐ 1011
- ☐ 9

30. Дан код.

```
int a= 10;  
int b= 11;  
String res = Integer.toHexString(a^b);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☒ 1
- ☐ a
- ☐ 1001

- 1011
- b

31. Дан код.

```
int a= 10;  
String res = Integer.toHexString(a<<1);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- b
- a
- 10
- 20
- Ⓐ 14

32. К стилям (парадигмам) программирования относят:

- ☒ императивный
- ☒ функциональный
- ☐ итерационный
- ☐ линейный

33. К стилям (парадигмам) программирования относят:

- ☒ объектно-ориентированный
- ☒ процедурный (императивный)
- ☐ ветвящийся
- ☐ многопоточный

34. Основным преимуществом объектно-ориентированного программирования перед императивным является

- Ⓐ сокращение количества межмодульных вызовов и уменьшение объемов информации, передаваемой между модулями
- отказ от указателей
- использование специальных библиотек
- повышенные требования к аппаратному обеспечению

35. Статический аспект объектно-ориентированной декомпозиции программной системы предполагает наличие следующих отношений между классами

- ☒ ассоциация
- ☒ зависимость
- ☐ сцепление
- ☐ делегирование

36. К принципам ООП относят

- ☒ иерархия
- ☒ инкапсуляция
- ☐ динамическое распределение памяти
- ☐ переопределение методов



37. К принципам ООП относят

- ☒ полиморфизм
- ☒ инкапсуляция
- ☐ статическое распределение памяти
- ☐ перегрузка методов

38. К принципам ООП относят

- ☒ иерархия
- ☒ абстрагирование
- ☐ применение интерфейсов
- ☐ применение статических членов класса

39. К принципам ООП относят

- ☒ модульность
- ☒ иерархия
- ☐ динамическое распределение памяти
- ☐ приведение типов

40. К принципам ООП относят

- ☒ типизация
- ☒ параллелизм
- ☐ статическое распределение памяти
- ☐ использование указателей

### 7.3. Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

#### 7.3.1. Вопросы к промежуточной аттестации

Семестр \_\_\_\_\_ 6 \_\_\_\_\_

№ п/п	Вопросы к экзамену
1	Понятие технологии Java: общие сведения, состав (язык программирования и платформа Java).
2	Процесс разработки Java SE приложения. Принципы обеспечения кроссплатформенности Java приложений.
3	JVM: место в платформе Java, области памяти.
4	JVM: управление памятью, острова изоляции.
5	Загрузка классов в Java.
6	JDK: назначение, установка, основные утилиты JDK.
7	JDK: Работа с jar файлами. Файлы манифеста. Entry Point.
8	Структура программы на Java.
9	Лексика языка Java: Unicode, комментарии, Java doc комментарии
10	Лексика языка Java: идентификаторы, целочисленные литералы.
11	Лексика языка Java: вещественные и логические литералы.
12	Лексика языка Java: строковые и символьные литералы, escape последовательности.
13	Java Code conventions: основные положения.
14	Характеристика языка Java с точки зрения типизации. Классификация типов в Java.
15	Переменные и константы в Java. Виды переменных Java и соответствующие области памяти JVM.
16	Простые и квалифицированные имена.
17	Операции: понятие и классификация по числу операндов.
18	Операция присваивание и арифметические операции.
19	Логические операции.
20	Приведение типов в операциях (явное и неявное). Проблемы
21	Выражения: понятие, правила вычисления.
22	Операторы Java: понятие, классификация.
23	Операторы ветвлений и циклов.
24	Аннотации Java.
25	Массивы: понятие, объявление, инициализация, способы создания.
26	Работа с массивами Java: доступ к элементам. Члены типа Array.
27	Клонирование одномерных и многомерных массивов.
28	Класс Arrays.
29	Перечисления в Java.
30	Область видимости имен.

№ п/п	Вопросы к экзамену
31	Принципы ООП.
32	Основные характеристика языка Java как ОО языка программирования. Объектная модель Java.
33	Объявление классов. Модификаторы доступа. Члены класса.
34	Конструкторы. Создание экземпляров класса
35	Методы класса.
36	Абстрактные методы и классы.
37	Окончательные члены и классы.
38	Статические члены класса.
39	Методы с переменным числом параметров.
40	Инкапсуляция и наследования в Java. Отношения «HAS-A» и «IS-A».
41	Сигнатура метода. Полиморфизм. Перегрузка и переопределение методов. Аннотация Override.
42	Вложенные классы.
43	Анонимные классы.
44	Пакеты: понятие, назначение, объявление, импорт.
45	Интерфейсы Java: понятие, назначение, объявление, модификаторы, реализация.
46	Интерфейсы Java 8.
47	Проектирование классов и интерфейсов в UML (изображения классов и интерфейсов, отношения между классами/интерфейсами) и отображение диаграмм UML в код Java.
48	Class Object: назначение, обзор методов. Контракт и реализация метода equals().
49	Class Object: обзор методов. контракт и реализация метода hashCode().
50	Mutable и Immutable классы.
51	Class Class.
52	Wrapper Classes.
53	Автоматическая упаковка и распаковка.
54	Класс BigInteger.
55	Класс BigDecimal.
56	Generics: типы, методы, ограниченные (bounded) типы.
57	Классы для работы со строками. Интерфейс CharSequence.
58	Класс String: назначение, особенности, конструкторы, создание строк.
59	Форматирование строк. Поиск символов и подстрок в строке.
60	Замена и сравнение строк.
61	Класс StringBuilder.
62	Класс StringTokenizer.
63	Основы регулярных выражений.
64	Интернационализация и локализация. Класс ResourceBundle.
65	Интернационализация и локализация. Класс Locale.

<b>№ п/п</b>	<b>Вопросы к экзамену</b>
66	Работа с датой и временем в Java 7: класс Date.
67	Работа с датой и временем в Java 7: класс Calendar.
68	Работа с датой и временем в Java 7: форматирование даты и времени.
69	Принципы работы с датой-временем в Java 8.
70	Работа с датой-временем средствами java.time (Java 8).
71	Понятие коллекции. Преимущества. Элементы Java Collections Framework.
72	Java Collections Framework: основные интерфейсы.
73	Java Collections Framework: основные реализации.
74	Интерфейс Collection: назначение, особенности, основные методы, об-ход коллекции.
75	Интерфейс Set: назначение, особенности, основные методы.
76	Интерфейс List: назначение, особенности, основные методы.
77	Интерфейс Queue: назначение, особенности, основные методы.
78	Интерфейс Map: назначение, особенности, основные методы.
79	Сортировка элементов в коллекциях.
80	Класс Collections.
81	Класс Arrays.
82	Виды реализаций Java Collection Framework. Обзор General-purpose реализаций.
83	Состав Java I/O: возможности пакетов java.io и java.nio.
84	Понятие потока. Байтовые и символьные потоки.
85	Понятие потока. Входные и выходные потоки.
86	Буферизированные потоки. Общий алгоритм работы с потоками.
87	Иерархия символьных потоков: обзор.
88	Иерархия байтовых потоков: обзор.
89	Классы, создающие потоки.
90	Классы, управляющие потоками.
91	Цепочки потоков и их использование в Java I/O.
92	Потоки преобразования байтов в примитивные типы.
93	Вывод на строчные устройства.
94	Потоки класса System
95	Класс StreamTokenizer.
96	Класс SequenceInputStream.
97	Класс LineNumberReader.
98	Класс RandomAccessFile.
99	Класс Console.
100	Управление жизненным циклом потока с использованием try-with- recourse.
101	Класс FileSystems.
102	Интерфейс Path и класс Paths.

№ п/п	Вопросы к экзамену
103	Класс Files.
104	Работа с Random Access Files
105	Понятие исключения. Механизм перехвата и обработки исключения. Exception object.
106	The Catch or Specify Requirement.
107	Классификация и иерархия исключений на Java.
108	Класс Throwable.
109	Обработка исключений. Try-with-resources.
110	Выбрасывание исключений. Цепочки исключений.
111	Logging API.
112	Потоки и процессы. Стандарт Posix и области памяти JVM.
113	Класс Thread и интерфейс Runnable. Способы создания потоков.
114	Приставка и прерывание спящего потока.
115	Методы wait() и notify(). Общий алгоритм работы на мониторе в Java.
116	Потоки-пользователи и потоки-демоны.
117	Правило <i>happens-before</i> .
118	Синхронизация. Критические секции.
119	Atomic.
120	volatile.
121	Проблемы многопоточности и рекомендации по их обходу в Java.
122	Блокировки.
123	Executors: основные понятия.
124	Thread Pools
125	Fork/join framework.
126	Потокобезопасные коллекции.
127	Понятие GUI. Основные элементы GUI.
128	Обзор Java технологий для создания GUI.
129	Компоненты Swing.
130	Применение MVC в Swing.
131	Компоненты JFC.
132	Понятие XML. Структура и правила XML документов.
133	Элементы XML.
134	Понятие схемы документа XML. DTD.
135	Понятие схемы документа XML. XSD.
136	Программные интерфейсы для работы с XML.
137	Алгоритм и виды работы парсеров.
138	DOM парсер: понятие, узлы.
139	Методы DOM парсера.

<b>№ п/п</b>	<b>Вопросы к экзамену</b>
140	SAX парсер: понятие, события.
141	Построение приложения с SAX парсером.
142	JAXP: понятие, обзор технологий.
143	JAXB: понятие, обзор технологий.
144	JDBC: понятие, структура пакетов, набор компонентов.
145	Использование JDBC в двухуровневых и многоуровневых приложениях.
146	Категории JDBC драйверов.
147	java.sql.DriverManager.
148	java.sql.Connection.
149	java.sql.Statement.
150	java.sql.ResultSet.
151	Алгоритм работы с JDBC.
152	PreparedStatement и CallableStatement.
153	Обработка SQLExceptions.
154	Обзор технологии RMI.
155	Создание удаленного RMI объекта.
156	Создание RMI клиента.

### 7.3.2. Критерии и нормы оценки

<b>Семестр</b>	<b>Форма проведения промежуточной аттестации</b>	<b>Критерии и нормы оценки</b>	
6	Экзамен (по накопительному рейтингу)	«отлично»	рейтинговый балл 85-100
		«хорошо»	рейтинговый балл 70-84
		«удовлетворительно»	рейтинговый балл 55-69
		«неудовлетворительно»	рейтинговый балл 0-54

## 8. Учебно-методическое и информационное обеспечение дисциплины

### 8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Курбатова, И. В.	Основы программирования на языке Java : учебное пособие для вузов / И. В. Курбатова, А. В. Печуров. — 2-е изд., стер. — Санкт-Петербург : Лань, 2026. — 348 с. — ISBN 978-5-507-51392-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <a href="https://e.lanbook.com/book/511511">https://e.lanbook.com/book/511511</a>	учебное пособие	2026	ЭБС «Лань»
2	Шнейдеров, Е. Н.	Разработка приложений на языке Java. Лабораторный практикум : учебное пособие / Е. Н. Шнейдеров, А. Ю. Писарчик, В. О. Казючиц. — Минск : БГУИР, 2023. — 92 с. — ISBN 978-985-543-561-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <a href="https://e.lanbook.com/book/479516">https://e.lanbook.com/book/479516</a>	лабораторный практикум	2023	ЭБС «Лань»
3	Вязовик, Н. А.	Вязовик, Н. А. Программирование на Java : учебное пособие / Н. А. Вязовик. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 601 с. — ISBN 978-5-4497-0852-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <a href="https://www.iprbookshop.ru/146383.html">https://www.iprbookshop.ru/146383.html</a>	учебное пособие	2025	ЭБС «IPRbooks»

<b>№ п/п</b>	<b>Авторы, составители</b>	<b>Заглавие (заголовок)</b>	<b>Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)</b>	<b>Год издания</b>	<b>Количество в научной библиотеке / Наименование ЭБС</b>
4	Блох, Дж.	Java. Эффективное программирование / Дж. Блох ; перевод В. Стрельцов ; под редакцией Р. Усманова. — 3-е изд. — Саратов : Профобразование, 2024. — 310 с. — ISBN 978-5-4488-0127-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <a href="https://www.iprbookshop.ru/145890.html">https://www.iprbookshop.ru/145890.html</a>	учебник	2024	ЭБС «IPRbooks»

## 8.2. Дополнительная литература

<b>№ п/п</b>	<b>Авторы, составители</b>	<b>Заглавие (заголовок)</b>	<b>Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)</b>	<b>Год издания</b>	<b>Количество в научной библиотеке / Наименование ЭБС</b>
1	Васюткина И.А.	Технология разработки объектно-ориентированных программ на JAVA [Электронный ресурс]: учебно-методическое пособие/ И.А. Васюткина. — Электрон. текстовые данные. — Новосибирск: Новосибирский государственный технический университет, 2012. — 152 с.— 978-5-7782-1973-1.	Учебно- методическое пособие	2012	ЭБС «IPRbooks»
2	Свиштунов А.Н.	Построение распределенных систем на Java [Электронный ресурс] / А.Н. Свиштунов. — 2-е изд. — Электрон. текстовые данные.— М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 317 с.	учебное пособие	2016	ЭБС «IPRbooks»



<b>№ п/п</b>	<b>Авторы, составители</b>	<b>Заглавие (заголовок)</b>	<b>Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)</b>	<b>Год издания</b>	<b>Количество в научной библиотеке / Наименование ЭБС</b>
		— 2227-8397. — Режим доступа: <a href="http://www.iprbookshop.ru/73707.html">http://www.iprbookshop.ru/73707.html</a>			
3	Монажв В.В.	Язык программирования Java и среда NetBeans [Электронный ресурс] / В.В. Монажв. — 2-е изд. — Электрон. текстовые данные.— М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 450 с. — 2227-8397. — Режим доступа: <a href="http://www.iprbookshop.ru/73739.html">http://www.iprbookshop.ru/73739.html</a>	учебное пособие	2016	ЭБС «IPRbooks»
4	Николаев Е.И.	Объектно-ориентированное программирование. Часть 1 [Электронный ресурс] : лабораторный практикум / Е.И. Николаев.— Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015. — 183 с. — 2227-8397. — Режим доступа: <a href="http://www.iprbookshop.ru/62966.html">http://www.iprbookshop.ru/62966.html</a>	лабораторный практикум	2015	ЭБС «IPRbooks»

### 8.3. Перечень профессиональных баз данных и информационных справочных систем

1. Oracle Java Documentation : официальная документация платформы Java SE и JDK. – Режим доступа: <https://docs.oracle.com/javase/>
2. Java SE API Documentation : справочная система по пакетам, классам и методам Java SE. – Режим доступа: <https://docs.oracle.com/en/java/javase/>
3. Dev.java : официальный образовательный ресурс для изучения Java и инструментов разработки. – Режим доступа: <https://dev.java/>
4. Java Community Process (JCP) : ресурс, содержащий сведения о стандартах и спецификациях платформы Java. – Режим доступа: <https://www.jcp.org/>

### 8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты (дата, номер, срок действия)
1	Visual Paradigm Community Edition	Бессрочно <a href="https://www.visual-paradigm.com/download/community.jsp">https://www.visual-paradigm.com/download/community.jsp</a>
2	Java SE Development Kit (JDK)	Бессрочно <a href="https://www.oracle.com/java/technologies/downloads/">https://www.oracle.com/java/technologies/downloads/</a>
3	Apache NetBeans IDE	Бессрочно <a href="https://netbeans.apache.org/download/index.html">https://netbeans.apache.org/download/index.html</a>
4	Eclipse IDE for Java Developers	Бессрочно <a href="https://www.eclipse.org/downloads/">https://www.eclipse.org/downloads/</a>
5	IntelliJ IDEA Community Edition	Бессрочно <a href="https://www.jetbrains.com/idea/download/">https://www.jetbrains.com/idea/download/</a>

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

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
1	Компьютерный класс. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для проведения лабораторных работ. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (УЛК-401).	Компьютер (монитор 19", системный блок Pentium (R) Dual-Core E5500 2,8 GHz / 4 Gb / 500 Gb), столы ученические , столы компьютерные , стол преподавательский, стулья, доска аудиторная(меловая)
2	Компьютерный класс. Учебная аудитория для проведения занятий лекционного типа.	Системные блоки (HP ProDesk), мониторы (Samsung), коммутатор (D-Link),

№ п/п	<b>Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)</b>	<b>Перечень основного оборудования</b>
	Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для проведения лабораторных работ. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (УЛК-402).	столы ученические, столы компьютерные, стулья, доска аудиторная.
3	Компьютерный класс. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для проведения лабораторных работ. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (УЛК-407).	Компьютер (монитор Samsung Sync Master 943n 19" , системный блок Intel (R) Core 2 Quad 2,40 GHz 1 Gb), столы лабораторные, стулья , доска 3-х секционная(меловая), стол преподавательский.
4	Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (УЛК-418).	Стол ученические двухместные (моноблок), доска аудиторная 3-х секционная (меловая), стол преподавательский , стулья, проектор Acer
5	Помещение для самостоятельной работы студентов (Г-401)	Стол, стулья, компьютеры
6	Помещение для самостоятельной работы обучающихся (С-401).	Шкафы для документации, доски магнитные, столы письменные, столы компьютерные