Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 02.11.2026 10:49:07
Уникальный программный ключ:
c3b3b9c625f6c113afa2a2c42baff9e05a38b76e

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования «Поволжский государственный университет сервиса»
(ФГБОУ ВО «ПВГУС»)

УТВЕРЖДЕНО

на заседании Высшей школы интеллектуальных систем и кибертехнологий

Протокол от 23.05.2025 № 10

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(фонд оценочных средств)
для проведения промежуточной аттестации по дисциплине

«Программирование»
наименование дисциплины

по образовательной программе высшего образования – программе **бакалавриат**

«Информационные системы и технологии»
наименование образовательной программы

09.03.02 «Информационные системы и технологии»
шифр, наименование направления подготовки / специальности

Составители Коновалова Анжелика Денисовна, старший преподаватель, Высшая школа интеллектуальных систем и кибертехнологий
ФИО, должность, структурное подразделение, ученая степень, ученое звание

Тольятти
2025

1. Паспорт фонда оценочных средств (далее – ФОС)

1.1. Планируемые результаты обучения по дисциплине

Код компетенции	Наименование компетенции	Код и наименование индикаторов достижения компетенции
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ИОПК-6.2. Использует современные языки программирования для разработки алгоритмов и программ ИОПК-6.3. Владеет методами отладки и тестирования программ
ПК-1	Способен к выполнению работ по проектированию программного обеспечения	ИПК-1.1. Осуществляет разработку требований к программному обеспечению и анализ исполнения требований ИПК-1.2. Выполняет разработку технических специфик

В результате изучения дисциплины обучающийся должен:

владеть навыками: навыками написания, запуска, отладки, тестирования и документирования программ, пригодных для практического применения

навыками декомпозиции требований, подготовки технических спецификаций и согласования требований с проектным решением

уметь: разрабатывать алгоритмы, реализовывать их на языке программирования, организовывать программу из функций и модулей, применять структуры данных, находить и исправлять ошибки, проектировать тестовые случаи

выявлять и формализовать функциональные и нефункциональные требования, анализировать их выполнимость и непротиворечивость, разрабатывать технические спецификации программных компонентов

знать: синтаксис и семантику основных конструкций современного языка программирования; способы представления алгоритмов; принципы модульной и объектно-ориентированной организации программ; методы отладки и тестирования

назначение и структуру требований к программному обеспечению, способы их анализа и трассировки; назначение технической спецификации и критериев приемки

1.2. Содержание дисциплины

№	Тема (раздел дисциплины) (в соответствии с РПД)	Код компетенции
1.	ВВЕДЕНИЕ В JAVA. СРЕДА РАЗРАБОТКИ. БАЗОВЫЙ СИНТАКСИС, ТИПЫ ДАННЫХ И ОПЕРАЦИИ	ОПК-6 ПК-1
2.	УПРАВЛЯЮЩИЕ КОНСТРУКЦИИ. УСЛОВНЫЕ ОПЕРАТОРЫ И ЛОГИКА В JAVA	ОПК-6 ПК-1
3.	ЦИКЛЫ В JAVA. ПОСЛЕДОВАТЕЛЬНОСТИ И НАКОПИТЕЛЬНЫЕ ВЫЧИСЛЕНИЯ	ОПК-6 ПК-1
4.	МАССИВЫ, МАТРИЦЫ И СТРОКИ В JAVA	ОПК-6 ПК-1
5.	МЕТОДЫ (ФУНКЦИИ) И МОДУЛЬНАЯ ОРГАНИЗАЦИЯ ПРОГРАММЫ В JAVA	ОПК-6 ПК-1
6.	РАБОТА С ФАЙЛАМИ, СТРУКТУРАМИ ДАННЫХ И ИСКЛЮЧЕНИЯМИ В JAVA. ОСНОВЫ ОТЛАДКИ И ТЕСТИРОВАНИЯ	ОПК-6 ПК-1
7.	РЕКУРСИЯ, ДИНАМИЧЕСКИЕ СТРУКТУРЫ ДАННЫХ И ОЦЕНКА АЛГОРИТМОВ В JAVA	ОПК-6 ПК-1
8.	ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ В JAVA: КЛАССЫ И ОБЪЕКТЫ	ОПК-6 ПК-1
9.	НАСЛЕДОВАНИЕ, ПОЛИМОРФИЗМ И ИНТЕРФЕЙСЫ В JAVA	ОПК-6 ПК-1
10.	КОЛЛЕКЦИИ, ОБОБЩЕНИЯ И ФУНКЦИОНАЛЬНЫЕ СРЕДСТВА JAVA	ОПК-6 ПК-1
11.	ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ И ТЕХНИЧЕСКИЕ СПЕЦИФИКАЦИИ	ОПК-6 ПК-1
12.	КОМПЛЕКСНАЯ РАЗРАБОТКА JAVA-ПРИЛОЖЕНИЯ: АРХИТЕКТУРА, ТЕСТИРОВАНИЕ, ДОКУМЕНТИРОВАНИЕ	ОПК-6 ПК-1

1.3. Система оценивания по дисциплине

Дисциплина изучается в течение двух семестров.

Форма промежуточной аттестации по дисциплине – зачет/экзамен. /КР

Уровень сформированности	100-балльная шкала, %	Шкала освоения, %	5-балльная дифференцированная оценка/балл	Недифференцированная оценка
допороговый	ниже 61	ниже 61	«неудовлетворительно» / 2	не зачтено
пороговый	61–85,9	61–69,9	«удовлетворительно» / 3	зачтено
пороговый	61–85,9	70–85,9	«хорошо» / 4	зачтено
повышенный	86–100	86–100	«отлично» / 5	зачтено

2. Перечень оценочных материалов для проведения промежуточной аттестации

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в ходе текущего контроля успеваемости (в процессе проведения практических занятий, тестирования, опросов).

В ходе проведения промежуточной аттестации осуществляется контроль и оценка результатов освоения компетенций.

Перечень вопросов к защите курсового проекта /работы

1. Архитектура Java: JVM, JRE, JDK. Компиляция и интерпретация байт-кода.
2. Обзор сред разработки (IDE). Структура Java-программы: классы, метод main().
3. Базовые примитивные типы данных в Java. Ссылочные типы данных. Переменные и константы.
4. Операции в Java: арифметические, сравнения, логические, побитовые, тернарный оператор.
5. Преобразование и приведение типов данных в Java.
6. Ввод/вывод данных: класс Scanner, System.out.println(). Форматированный вывод.
7. Логический тип boolean. Операции сравнения и логические операторы (&&, ||, !).
8. Условный оператор if, if-else. Множественный выбор с использованием switch.
9. Вложенные ветвления. Проверка предусловий. Обработка некорректных входных данных.
10. Использование метода equals() для сравнения строк.
11. Циклы в Java: for, while, do-while. Управление циклом (операторы break, continue).
12. Счетчики и накопители в циклах. Организация накопительных вычислений.
13. Вложенные циклы. Генерация числовых последовательностей.
14. Циклы для обработки строк. Бесконечные циклы и контроль завершения. Итераторы.
15. Массивы в Java: одномерные и многомерные. Объявление, инициализация, заполнение.

Перечень вопросов и заданий для подготовки к зачету (3 семестр)

ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий

1. Архитектура Java: JVM, JRE, JDK. Компиляция и интерпретация байт-кода.
2. Обзор сред разработки (IDE). Структура Java-программы: классы, метод main().
3. Базовые примитивные типы данных в Java. Ссылочные типы данных. Переменные и константы.
4. Операции в Java: арифметические, сравнения, логические, побитовые, тернарный оператор.
5. Преобразование и приведение типов данных в Java.
6. Ввод/вывод данных: класс Scanner, System.out.println(). Форматированный вывод.
7. Логический тип boolean. Операции сравнения и логические операторы (&&, ||, !).
8. Условный оператор if, if-else. Множественный выбор с использованием switch.
9. Вложенные ветвления. Проверка предусловий. Обработка некорректных входных данных.
10. Использование метода equals() для сравнения строк.
11. Циклы в Java: for, while, do-while. Управление циклом (операторы break, continue).
12. Счетчики и накопители в циклах. Организация накопительных вычислений.
13. Вложенные циклы. Генерация числовых последовательностей.
14. Циклы для обработки строк. Бесконечные циклы и контроль завершения. Итераторы.
15. Массивы в Java: одномерные и многомерные. Объявление, инициализация, заполнение.
16. Индексация и обход массивов. Алгоритмы поиска элементов в массиве.
17. Алгоритмы сортировки массивов (базовые методы).
18. Строки в Java. Основные методы работы со строками. Сравнение строк.
19. Классы StringBuilder и StringBuffer для работы с изменяемыми строками.
20. Методы (функции) в Java. Объявление методов. Передача параметров по значению.
21. Возврат значений из методов. Перегрузка методов.
22. Область видимости переменных. Статические и нестатические методы.
23. Понятие рекурсивных методов. Принцип работы рекурсии. Стек вызовов.

24. Организация пакетов и импорт классов. Модификаторы доступа (public, private, protected).
25. Принципы написания чистого кода в Java: именование, длина методов, комментарии.
26. Пользовательские типы данных в Java. Структуры данных.
27. Работа с текстовыми файлами. Работа с двоичными файлами.
28. Иерархия исключений в Java. Проверяемые и непроверяемые исключения.
29. Конструкция try-catch-finally. Ресурсный блок try-with-resources.
30. Создание собственных исключений.
31. Основы отладки в IntelliJ IDEA: точки останова, пошаговое выполнение, просмотр переменных.
32. Этапы разработки программы: постановка задачи, формализация, алгоритмизация, кодирование, тестирование.
33. Принципы тестирования программы. Тестовые случаи и регрессионная проверка.

ПК-1. Способен к выполнению работ по проектированию программного обеспечения

34. Понятие требования к программному обеспечению. Источники требований.
35. Функциональные и нефункциональные требования. Критерии качества требований.
36. Анализ реализуемости требований в контексте Java-стек технологий.
37. Декомпозиция требований на модули. Техническая спецификация программного компонента.
38. Назначение, интерфейсы, структуры данных, ограничения, критерии приемки в технической спецификации.
39. Критерии приемки и их связь с тест-кейсами.
40. Связь требований, архитектурных решений и исходного кода.

Перечень вопросов и заданий для подготовки к экзамену (4 семестр)

ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий

1. Архитектура Java: JVM, JRE, JDK. Компиляция и интерпретация байт-кода. Структура Java-программы.
2. Базовые примитивные типы данных. Ссылочные типы. Преобразование и приведение типов.
3. Операции в Java: арифметические, сравнения, логические, побитовые, тернарный оператор.
4. Ввод/вывод данных: класс Scanner, System.out.println(). Форматированный вывод.
5. Условный оператор if, if-else. Множественный выбор с использованием switch. Вложенные ветвления.
6. Циклы в Java: for, while, do-while. Управление циклом. Вложенные циклы.
7. Массивы в Java: одномерные и многомерные. Объявление, инициализация, индексация и обход.
8. Алгоритмы поиска и сортировки массивов.
9. Строки в Java. Основные методы. Классы String, StringBuilder, StringBuffer.
10. Методы (функции) в Java. Объявление, параметры, возврат значений. Перегрузка методов.
11. Область видимости переменных. Статические и нестатические методы. Модификаторы доступа.
12. Понятие рекурсии. Рекурсивные методы. Стек вызовов и ограничения рекурсии.
13. Сравнение итеративных и рекурсивных решений.
14. Пользовательские типы данных. Структуры данных.
15. Работа с текстовыми и двоичными файлами. Иерархия исключений в Java.
16. Конструкция try-catch-finally. try-with-resources. Создание собственных исключений.
17. Основы ООП в Java: классы и объекты. Поля, методы, конструкторы.
18. Инкапсуляция: модификаторы доступа. Геттеры и сеттеры. Жизненный цикл объекта.
19. Ключевое слово this. Композиция и агрегация. Ассоциации между классами.
20. Принципы проектирования простых классов.
21. Наследование в Java. Повторное использование кода. Переопределение методов.
22. Полиморфизм. Абстрактные классы. Интерфейсы в Java.

23. Дефолтные и статические методы в интерфейсах. Множественное наследование через интерфейсы.
24. Принцип подстановки Лисков. Композиция vs наследование.
25. Инкапсуляция, наследование и полиморфизм — три кита ООП.
26. Java Collections Framework (JCF): иерархия. Интерфейсы Collection, List, Set, Queue, Map.
27. Классы-реализации: ArrayList, LinkedList, HashSet, TreeSet, HashMap, TreeMap.
28. Итераторы. Цикл for-each. Обобщенные типы: параметризованные классы и методы.
29. Ограничения обобщений. Сравнение структур данных по времени выполнения операций.
30. Функциональные интерфейсы. Лямбда-выражения. Stream API: фильтрация, сортировка, группировка.
31. Основы отладки в IntelliJ IDEA. Принципы тестирования программы. Тестовые случаи.
32. Сборка программного решения из модулей. Организация зависимостей с помощью Maven/Gradle.
33. Стратегии тестирования: модульное, интеграционное, регрессионное.
34. Документирование кода (Javadoc). Документирование пользовательских сценариев.
35. Упаковка приложения в JAR-файл. Подготовка программного решения к демонстрации и защите.

ПК-1. Способен к выполнению работ по проектированию программного обеспечения

36. Понятие требования к программному обеспечению. Источники требований. Функциональные и нефункциональные требования.
37. Критерии качества требований: однозначность, проверяемость, выполнимость, непротиворечивость.
38. Анализ реализуемости требований к программному обеспечению. Выявление конфликтующих и неполных требований.
39. Декомпозиция требований на функции программной системы. Трассируемость требований и ее назначение.
40. Критерии приемки и связь с тестированием.
41. Назначение технической спецификации программного компонента. Структура технической спецификации.
42. Спецификация интерфейсов программных модулей. Спецификация структур и форматов данных.
43. Описание исключительных ситуаций и ошибок в спецификации.
44. Связь требований, архитектурных решений и исходного кода.
45. Проверка программной реализации на соответствие требованиям. Рецензирование требований и технических спецификаций.
46. Использование прототипа для уточнения требований. Документирование изменений требований.

Типовые практические задания для зачета и экзамена

1. Разработать программу обработки последовательности/массива по заданному условию (поиск, сортировка, подсчет суммы и т.д.).
2. Разработать функцию или модуль с заданным интерфейсом и тестами.
3. Спроектировать класс предметной области и реализовать проверку его инвариантов (корректности состояния).
4. Найти и исправить дефект в предложенном фрагменте программы, описать тест, выявляющий ошибку.
5. По исходному описанию сформулировать функциональные требования и фрагмент технической спецификации.
6. Проверить соответствие фрагмента реализации заданным требованиям.
7. Разработать иерархию классов и продемонстрировать полиморфное поведение.
8. Обработать коллекцию объектов с использованием Stream API (фильтрация, сортировка, группировка).