

## Список вопросов и задач для экзамена по дисциплине «Современные технологии программирования»

Летняя сессия 22/23 уч. года

### Список вопросов

1. Библиотека Swing, общие черты и особенности.
2. Виды контейнеров в Swing.
3. Элементы пользовательского интерфейса Swing.
4. Модель событий Swing. Интерфейс `EventListener`.
5. Менеджеры компоновки Swing.
6. `GUI Designer Swing`
7. Текстовые поля в Swing
8. Компонент управления `JButton` в Swing
9. Платформа `JavaFX`, особенности, компоненты
10. Шаблон MVC (Model-View-Controller) в Spring.
11. Классы `StringBuffer` и `StringBuilder`.
12. Архитектура JDBC (Java DataBase Connectivity). Двух и трехуровневые модели доступа к базе данных. Преимущества и недостатки JDBC.
13. Массивы Java: объявление, инициализация. Основные методы класса `Arrays`. Доступ к элементам массивов, итерация массивов. Двумерные массивы.
14. Иерархия наследования Java. Преобразование типов при наследовании. Ключевое слово `instanceof`.
15. Интерфейсы Java: определение интерфейса, реализация интерфейса. Преимущества применения интерфейсов. Переменные интерфейсов. Наследование интерфейсов. Методы по умолчанию. Статические методы интерфейсов.
16. Байтовые потоки `InputStream` и `OutputStream`. Консольный ввод и вывод Java. Символьные потоки данных. Абстрактные классы `Writer`, `Reader`.
17. Основные фреймворки и задачи, решаемые Spring.
18. Spring Inversion of Control (IoC) контейнер Spring.
19. Dependency Injection (DI) в Spring.
20. Жизненный цикл объекта Bean Spring.
21. Конфигурация `ApplicationContext` с помощью xml в Spring.
22. Область видимости Bean в Spring.
23. Фабричные или `factory`-методы в Spring.
24. Конфигурация `ApplicationContext` с помощью аннотаций в Spring.
25. Связывание в Spring, аннотация `@Autowired`.
26. Архитектурный стиль REST.
27. Spring Web-MVC, основная схема и логика работы.
28. Класс `DispatcherServlet`, его функции.
29. Маппинг в Spring.
30. Интерфейсы `HttpServletRequest` и `HttpServletResponse`.
31. Архитектурный стиль CRUD, его соответствие REST и HTTP.
32. Шаблон Data Access Object (DAO).
33. Основные понятия Объектно-реляционного отображения (ORM - Object-Relational Mapping).
34. Спецификация Java Persistence API (JPA).
35. Архитектура ORM Java Persistence API (JPA).
36. Основные аннотации Java Persistence API (JPA).
37. Библиотека `Hibernate`, основные аннотации.
38. Объявление сущности и таблицы в `Hibernate`.

39. Интерфейс Session в Hibernate.
40. Ассоциация сущностей в Hibernate.
41. Spring Boot: определение, характеристики, преимущества.
42. Spring Initializr, особенности и преимущества применения.
43. Структура фреймворка JUnit.
44. JUnit аннотации @Test, @DisplayName.
45. JUnit аннотации @BeforeEach, @AfterEach.
46. Тестовые классы и методы JUnit.
47. Утверждения JUnit. Класс Assert.
48. Тестирование исключений JUnit.
49. Генератор документирования Javadoc. Виды комментариев.
50. Дескрипторы Javadoc.

### *Задачи*

1. Условие: «Реализовать программу для выполнения следующих математических операций с целочисленным, байтовым и вещественным типами данных: сложение, вычитание, умножение, деление, деление по модулю (остаток), модуль числа, возведение в степень. Все данные вводятся с клавиатуры (класс Scanner, System.in, nextInt()).» По данному условию необходимо реализовать программу с интерактивным консольным меню, (т.е. вывод списка действий по цифрам. При этом при нажатии на цифру у нас должно выполняться определенное действие). При этом в программе данные пункты должны называться следующим образом:

1. Вывести все таблицы из MySQL.
2. Создать таблицу в MySQL.
3. Сложение чисел, результат сохранить в MySQL с последующим выводом в консоль.
4. Вычитание чисел, результат сохранить в MySQL с последующим выводом в консоль.
5. Умножение чисел, результат сохранить в MySQL с последующим выводом в консоль.
6. Деление чисел, результат сохранить в MySQL с последующим выводом в консоль.
7. Деление чисел по модулю (остаток), результат сохранить в MySQL с последующим выводом в консоль.
8. Возведение числа в модуль, результат сохранить в MySQL с последующим выводом в консоль.
9. Возведение числа в степень, результат сохранить в MySQL с последующим выводом в консоль.
10. Сохранить все данные (вышеполученные результаты) из MySQL в Excel и вывести на экран.

2. Напишите программу, в которой из строки "I have 3 cats, 4 dogs, and 1 turtle" отбираются цифры. Из этих цифр формируется массив.

3. Разработайте программу, которая выводит в консоль все цифры, входящие в натуральное число n. К примеру, если дано число 2359, то в консоль выводятся отдельно числа 2, 3, 5, 9.

4. Написать калькулятор для строковых выражений вида "<число> <операция> <число>", где <число> - положительное целое число меньше 10, записанное словами, например, "четыре", <арифметическая операция> - одна из операций "плюс", "минус", "умножить". Результат выполнения операции вернуть в виде текстового представления числа. Пример: "пять плюс четыре" --> "девять".

5. Напишите программную реализацию бинарного дерева поиска.

6. Разработайте программу, которая выводит буквы английского алфавита, используя цикл `while`.

7. Напишите программу, которая будет выводить в консоль введенное слово 6 раз.

8. Разработать программу для вывода на экран кубов первых десяти положительных чисел.

9. Напишите программу, которая по дате определяет день недели, на который эта дата приходится.

10. Написать класс, который при введении даты в формате ДД.ММ.ГГ (к примеру, 22.10.20) выводит номер недели. Даты начиная с 2020 по 2022 годы. К примеру, первая неделя в 2020 году: 1-5 января, вторая неделя – 6-12 января. Значит при вводе 08.01.20 вывод должен быть: Неделя 2.

11. Разработайте программу, реализующую рекурсивное вычисление факториала.

12. Разработать класс-оболочку для числового типа `double`. Реализовать статические методы сложения, деления, возведения в степень.

13. Разработать программу, которая заполняет двумерный массив случайными положительными числами в диапазоне от 1 до 100 до тех пор, пока сумма граничных элементов не станет равной 666. Пользователь вначале вводит размер матрицы.

14. Разработать программу, в которой требуется создать класс, описывающий геометрическую фигуру – треугольник. Методами класса должны быть – вычисление площади, периметра. Создать класс-наследник, определяющий прямоугольный треугольник.

15. Разработать программу, в которой требуется создать абстрактный класс. В этом абстрактном классе определить абстрактные методы вычисления функции в определенной точке. Создать классы-наследники абстрактного класса, описывающими уравнения прямой и параболы. Программа должна выводить в консоль значение функции при вводе определенного значения.

16. Создать интерфейс `Progress` с методами вычисления любого элемента прогрессии и суммы прогрессии. Разработать классы арифметической и геометрической прогрессии, которые имплементируют интерфейс `Progress`.

17. Разработать интерфейс `InArray`, в котором предусмотреть метод сложения двух массивов. Создать класс `ArraySum`, в котором имплементируется метод сложения массивов. Создать класс `OrArray`, в котором метод сложения массивов имплементируется как логическая операция ИЛИ между элементами массива.

18. Создать класс `Binary` для работы с двоичными числами фиксированной длины. Число должно быть массивом тип `char`, каждый элемент которого принимает значение 0 или 1. Младший бит имеет младший индекс. Отрицательные числа представляются в дополнительном коде. Дополнительный код получается инверсией всех битов с прибавлением 1 к младшему биту. Например, +1 – это в двоичном коде будет выглядеть, как 0000 0001. А -1 в двоичном коде будет выглядеть, как 1111 1110 + 0000 0001 = 1111 1111. Создать методы конвертации десятичного числа в массив и обратно.

19. Создать класс `Matrix` для работы с двумерными матрицами. Создать методы для генерации нулевой матрицы, а также для генерации матрицы со случайными величинами – применить `Math.random()`. Реализовать метод сложения матриц.

20. Реализовать класс `MyMath` для работы с числами. Реализовать статический метод класса `MyMath.round()`, который округляет дробь до целого числа. Также статический метод `abs()`, который находит модуль числа. Статический метод `MyMath.pow()` для нахождения степени числа. Библиотеку `Math` не использовать.

21. Разработать программу для игры «Угадайка». Программа загадывает случайное число от 1 до 10, требуется его отгадать с трех попыток. После каждой попытки, если результат неверен, игроку выводится сообщение, меньше или больше названное игроком число, чем загаданное. Сет заканчивается или если игрок угадывает число, или если исчерпывает три попытки, не угадав. Игра должна быть выполнена в бесконечном цикле, и продолжается до тех пор, пока на предложение «Сыграем еще раз?» игрок не напишет «Нет».

22. Разработайте программу-генератор рабочего календаря. Слесарь механосборочного цеха работает сутки через трое. Если смена попадает на воскресенье, то переносится на понедельник. По введенной дате программа должна генерировать расписание из дат на текущий месяц на 2022 год.

23. Разработать класс для представления комплексных чисел с возможностью задания вещественной и мнимой частей в виде массива из двух чисел типа `int`. Определить методы для выполнения операций сложения, вычитания и умножения комплексных чисел.

24. Создайте класс `Form` - оболочку для создания и ввода пароля. Он должен иметь методы `input`, `submit`, `password`. Создайте класс `SmartForm`, который будет наследовать от `Form` и сохранять значения `password`.

25. Сделайте класс `User`, в котором будут следующие `protected` поля - `name` (имя), `age` (возраст), `public` методы `setName`, `getName`, `setAge`, `getAge`. Сделайте класс `Worker`, который наследует от класса `User` и вносит дополнительное `private` поле `salary` (зарплата), а также методы `public` `getSalary` и `setSalary`. Создайте объект этого класса 'Иван', возраст 25, зарплата 1000. Создайте второй объект этого класса 'Вася', возраст 26, зарплата 2000. Найдите сумму зарплаты Ивана и Васи. Сделайте класс `Student`, который наследует от класса `User` и вносит дополнительные `private` поля стипендия, курс, а также геттеры и сеттеры для них.

26. Создайте класс `ColorModel` для определения цветовой модели. Разработайте подклассы `RGBconverter` и `CMYKconverter` для конвертации цвета из одной модели в другую. Конвертация CMYK в RGB производится по следующим формулам:  $R = 255 \times (1-C) \times (1-K)$ ,  $G = 255 \times (1-M) \times (1-K)$ ,  $B = 255 \times (1-Y) \times (1-K)$  (где R – red, G – green, B – black, C – Cyan, M - Magenta, Y - Yellow, K- Black))

27. Создайте класс `Number` для конвертации десятичного числа в бинарный, восьмеричный, шестнадцатеричный вид. Реализовать в виде статических методов класса. Числа вводятся с клавиатуры с запросом в какой численный вид конвертировать.

28. Разработать класс `Neuron` для реализации нейронной сети из двух нейронов и одного выхода. Сделать функцию прямого распространения с функцией активации в виде сигмоиды.

29. Напишите программу, которая заполняет списочный массив случайными числами типа Integer (значения этих чисел были от 1 до 100). Список должен содержать 100 элементов. Затем отсортируйте по убыванию список и выведите первые 10 значений в консоль.

30. Разработайте программу, которая заполняет список случайными числами. Количество элементов и числовой диапазон вводятся пользователем. Программа должна проверять, входит ли число (также вводится пользователем) в данный список. Поиск провести организовать, как бинарный.

31. На основе класса BitSet разработайте программу для реализации битовых операций AND, OR, XOR, а также маскирования.

32. Напишите программу, которая получает в качестве входных данных два числа. Эти числа являются количество строк и столбцов двумерной коллекции целых чисел. Далее элементы заполняются случайными числами и выводятся в консоль в виде таблицы.

33. Разработайте программу, которая получает в качестве параметра два числа – количество строк и столбцов двумерной коллекции целых чисел. Коллекция заполняется случайными числами, после чего на экран выводятся максимальное и минимальное значения с индексами ячеек.

34. Разработайте программу, в которой создайте две коллекции с именами людей (строковые переменные). Затем последовательно выводите в консоль имена.

35. Напишите программу, которая реализует класс Matrix и следующие методы:

- a. Сложение и вычитание матриц.
- b. Умножение матрицы на число.
- c. Произведение двух матриц.
- d. Транспонированная матрица.
- e. Возведение матрицы в степень.
- f. Если метод, возвращает матрицу, то он должен возвращать новый объект, а не менять базовый.

36. Разработать программу для поочередной обработки текстовых файлов. Файлы созданы со следующими именами: n.txt, где n – натуральное число. В файлах записаны: в первой строке одно число с плавающей запятой, во второй строке – второе число. Пользователь вводит название файла и требуемую операцию над числами (сложение, умножение, разность). Результат выводится на экран и файл n\_out.txt.

37. Написать приложение для сложения, вычитания, умножения, деления, возведения в степень логарифмов. Программа должна выполнять ввод данных, проверку правильности введенных данных, выдачу сообщений в случае ошибок. Результат выводится на экран и записывается в файл.

38. Разработать программу шифровки-дешифровки по алгоритму AES-128. Данные берутся из файла, зашифрованные данные сохраняются в указанный файл.

39. Разработать программу нахождения наибольшего общего делителя двух натуральных чисел. Требуется реализовать рекурсивный и без рекурсии варианты.

40. Напишите программу, которая каждые 5 секунд отображает на экране данные о времени, прошедшем от начала запуска программы, а другой её поток выводит сообщение

каждые 7 секунд. Третий поток выводит на экран сообщение каждые 10 секунд. Программа работает одну минуту, затем останавливается.

41. Условие задачи: «Ввести две строки (не менее 50 символов каждая) с клавиатуры. Необходимо вывести на экран две введенных ранее строки, подсчитать и вывести размер длины каждой строки, объединить данные строки в одну, сравнить данные строки и результат сравнения вывести на экран». По данному условию необходимо реализовать программу с интерактивным консольным меню, (т.е. вывод списка действий по цифрам. При этом при нажатии на цифру у нас должно выполняться определенное действие). При этом в программе данные пункты должны называться следующим образом:

1. Вывести все таблицы из MySQL.
2. Создать таблицу в MySQL.
3. Ввести две строки с клавиатуры, результат сохранить в MySQL с последующим выводом в консоль.
4. Подсчитать размер ранее введенных строк, результат сохранить в MySQL с последующим выводом в консоль.
5. Объединить две строки в единое целое, результат сохранить в MySQL с последующим выводом в консоль.
6. Сравнить две ранее введенные строки, результат сохранить в MySQL с последующим выводом в консоль.
7. Сохранить все данные (вышеполученные результаты) из MySQL в Excel и вывести на экран.

42. Написать на основе Spring Boot клиент-серверное приложение MyUser, в котором можно управлять данными пользователей из базы данных через веб-интерфейс: имя, фамилия, возраст, номер группы. База данных может быть любой – MySQL, PostgreSQL и т.д. При этом должна быть доступна возможность добавления/удаления/редактирования пользователей.

43. Написать на основе Spring Boot Security форму для авторизации и регистрации пользователя. При этом после авторизации пользователя должно быть перенаправление на главную страницу. Главная страница должна содержать запись «Hello World!». При этом до авторизации главная страница не должна быть доступна для пользователя.

44. Разработать MVC-приложение арифметический калькулятор на основе Spring Boot. Применить шаблонизатор Thymeleaf. Все результаты вычисления должны сохраняться и выводиться из MySQL.

45. Разработка веб-MVC приложения на основе Spring Boot. Приложение должно генерировать последовательность из 1000 случайных чисел в диапазоне, заданном пользователем, и выводит эти числа на экран и вычисляет их среднее арифметическое.

46. Разработать приложение для работы с локальной базой данных MySQL. Создайте базу данных мобильных телефонов (не менее 10 позиций), со следующими полями: производитель, модель, год выпуска, диагональ экрана. Напишите методы для выполнения запросов к базе данных. Все данные должны выводиться в консоли на экран.