

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

Б1.В.ДВ.01.01
(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование на Java (Джава) 1

(наименование дисциплины)

по направлению подготовки (специальности)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности в соответствии с ФГОС ВПО/ ФГОС ВО)

Цифровая трансформация бизнеса

(направленность (профиль)/специализация)

Форма обучения: очная

Год набора: 2020

Общая трудоемкость: 5 ЗЕ

Распределение часов дисциплины по семестрам

Семестр	4	Итого
Форма контроля	экзамен	
Вид занятий		
Лекции	34	34
Лабораторные		
Практические	34	34
Руководство: курсовые работы (проекты) / РГР		
Промежуточная аттестация	0,35	0,35
Контактная работа	68	68
Самостоятельная работа	76	76
Контроль	35,65	35,65
Итого	180	180

Рабочую программу составил(и):

Доцент, к.п.н., Ерофеева Елена Александровна

(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 (специальности)

09.03.03 Прикладная информатика

Срок действия рабочей программы дисциплины до «31» августа 2024 г.

УТВЕРЖДЕНО

На заседании кафедры

Прикладная математика и информатика

(протокол заседания № 1 от «09» сентября 2019г.).

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 и практики, на освоении которых базируется данная дисциплина: Компьютерные сети, Объектно-ориентированное программирование, Программирование систем компьютерной графики.

Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее: – Программирование на Java (Джава) 2, Производственная практика (технологическая (проектно-технологическая) практика), Производственная практика (преддипломная практика).

3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
ПК-6 Способен осуществлять выбор и моделирование решения для реализации программного обеспечения на языках программирования	ПК-6.1 Знает технологии моделирования программного обеспечения	Знать:технологии моделирования программного обеспечения Уметь:применять технологии моделирования программного обеспечения Владеть:навыками моделирования программного обеспечения
	ПК-6.2 Умеет осуществлять выбор и моделирование решения для реализации программного обеспечения на языках программирования	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-6.3 Владеет навыками выбора технологий моделирования решения для реализации программного обеспечения на языках программирования	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 (наименование оценочного средства)
1	Лек 1	Введение в Java	4	2		-	Тестовые задания
1	Пр3 1	Утилиты JDK	4	2	6	-	Отчет по практической работе
2	Лек 2	Основы синтаксиса Java.	4	2		-	Тестовые задания
2	Лек 3	Основы синтаксиса Java.	4	2		-	Тестовые задания
2	Пр3 2	Java Syntax 1	4	2	6	-	Отчет по практической работе
2	Пр3 3	Java Syntax 2	4	2	6	-	Отчет по практической работе
3	Лек 4	Объектно- ориентированное программирования в Java	4	2		-	Тестовые задания
3	Лек 5	Объектно- ориентированное программирования в Java	4	2		-	Тестовые задания
3	Пр3 4	Объектно- ориентированное программирования в Java 1	4	2	6	-	Отчет по практической работе
3	Пр3 5	Объектно- ориентированное программирования в Java 2	4	2	6	-	Отчет по практической работе
4	Лек 6	Интерфейсы	4	2		-	Тестовые задания
4	Лек 7	Интерфейсы	4	2		-	Тестовые задания
4	Пр3 6	Интерфейсы	4	2	6	-	Отчет по практической работе
4	Пр3 7	Интерфейсы	4	2	6	-	Отчет по практической работе
2	Лек 8	Обработка строк на Java	4	2		-	Тестовые задания
2	Лек 9	Обработка строк на Java	4	2		-	Тестовые задания
2	Пр3 8	Обработка строк на Java 1	4	2	6	-	Отчет по практической работе
2	Пр3 9	Обработка строк на Java 2	4	2	6	-	Отчет по практической работе
2	Лек 10	Работа с датой и временем на Java	4	2		-	Тестовые задания

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив , ч.	Формы текущего контроля (наименование оценочного средства)
2	Лек 11	Работа с датой и временем на Java	4	2		-	Тестовые задания
2	Пр3 10	Работа с датой и временем на Java	4	2	6	-	Отчет по практической работе
2	Пр3 11	Работа с датой и временем на Java	4	2	6	-	Отчет по практической работе
2	Лек 12	Обобщенные типы.	4	2		-	Тестовые задания
2	Лек 13	Обобщенные типы.	4	2		-	Тестовые задания
2	Пр3 12	Обобщенные типы	4	2	6	-	Отчет по практической работе
2	Пр3 13	Обобщенные типы	4	2	6	-	Отчет по практической работе
2	Лек 14	Java Collection Framework 1	4	2		-	Тестовые задания
2	Лек 15	Java Collection Framework 1	4	2		-	Тестовые задания
2	Пр3 14	Java Collection Framework 1	4	2	8	-	Отчет по практической работе
2	Пр3 15	Java Collection Framework 1	4	2	8	-	Отчет по практической работе
2	Лек 16	Java Collection Framework 2	4	2		-	Тестовые задания
2	Лек 17	Java Collection Framework 2	4	2		-	Тестовые задания
2	Пр3 16	Java Collection Framework 2	4	2	6	-	Отчет по практической работе
	Сам 1	Работа с теоретическим материалом	4	76			
	ПА	Промежуточная аттестация	4	0,35			
	Пр3 17 ТИ	Итоговый тест по курсу через ОТ (100 баллов)	4	2	100		
Итого:				180	100		

Схема расчета итогового балла

Текущий рейтинг (все занятия и Пр3) + Промежуточные тесты) + Результат итогового теста и все делится на 2

5. Образовательные технологии

В рамках изучения дисциплины предусмотрено использование следующих образовательных технологий:

- технология традиционного обучения;
- интерактивные технологии: учебные дискуссии (применяются во всех модулях по итогам выполнения работ).

Технологии традиционного обучения - организация учебного процесса в вузе, основанная на лекционных и практических формах обучения: объяснительно-иллюстративное обучение. Данная технология применяется во всех модулях курса.

Технология интерактивного обучения - организация учебного процесса, которая предполагает максимальную активность студентов в процессе формирования ключевых компетенций. На учебной дискуссии студенты представляют результат выполнения заданной работы. Проводится дискуссия по применённым решениям, обсуждается эффективность и архитектура кода.

6. Методические указания по освоению дисциплины

6.1. Рекомендации по подготовке к практическим занятиям

Студентам следует:

- при подготовке к занятиям обязательно использовать не только 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.

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.

6.2. 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
4	ПК-6	Тестовые задания Вопросы к экзамену Отчеты по практическим занятиям

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

7.2.1. Примеры тестовых заданий

(наименование оценочного средства)

Типовые примеры заданий

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- программ - 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- программ - 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 $a \& \& (b \parallel c) \wedge d$ 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 $a \& \& b \parallel c \wedge d$ 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

Краткое описание и регламент выполнения

К тестам допускаются все студенты.

По результатам итогового теста студент может набрать максимально 100 баллов.

7.2.2. Комплект отчетов по Практическим работам (Примеры)

(наименование оценочного средства)

Практическая работа №1 «Основные утилиты Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №2 «Основы синтаксиса Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №3 «Объектно-ориентированное Программирование в Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №4 «Интерфейсы»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №5 «Обработка строк на Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;

- выводы по работе.

Практическая работа №6 «Работа с датой и временем на Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №7 «Обобщенные типы»

Форма отчета по Практическому занятию №7

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №8 «Java Collection Framework»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №9 «Java Collection Framework»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Требования к оформлению

Отчет должен содержать подробное описание (включая иллюстративный материал) последовательности действий, сделанных студентом 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 студентом поставленной цели
6. Обоснованность применяемого решения
7. Грамотность (содержательная) используемых формулировок

Критерии оценки за отчеты по практическим работам:

Полностью выполненное и вовремя защищенный отчет – максимальный балл. За каждое невыполненное задание снимаются баллы в соответствии с заданием на практическое занятие. Просрочка на 1 неделю - коэффициент 0,75, за две - 0,5, за три - 0,25, за четыре и более - 0 (учитывается факт сдачи).

7.2.3. Комплект заданий для оценки сформированности компетенций (Примеры)

(наименование оценочного средства)

ПК-6 Способен осуществлять выбор и моделирование решения для реализации программного обеспечения на языках программирования

код и наименование компетенции

ОМ закрытого типа

Задание 1

Выберите один правильный вариант ответа.

Дан код. Какое действие описывает код?

```
import java.io.File;
public class Program {
    public static void main(String[] args) {
        File dir = new File("C://SomeDir");
        if(dir.isDirectory()){
            for(File item : dir.listFiles()){
                if(item.isDirectory()){
                    System.out.println(item.getName() + " \t folder");
                }
            }
        }
        else{
            System.out.println(item.getName() + "\t file");
        }
    }
}
```

- a. Запись текста в файл из определенного каталога
- b. Чтение текста из файла внутри каталога
- c. Получение всех подкаталогов и файлов в определенном каталоге
- d. Получение всех файлов в каталоге

Правильный ответ: c

Задание 2

Выберите один правильный вариант ответа.

Дан фрагмент кода. Каким будет результат?

```
private static String fileName = "C://blog/a.txt";
public static void main(String[] args)
    throws FileNotFoundException {
    FileWorker.delete(fileName);
}
```

- a. Создание файла
- b. Определение объекта для каталога
- c. Переименование файла
- d. Удаление файла

Правильный ответ: d

Задание 3

Выберите несколько правильных вариантов ответа.

Дан код. Какое действие описывает код?

```
import java.io.*;
public class Program {
    public static void main(String[] args) {
        String text = "Hello world!";
        try(FileOutputStream fos=new FileOutputStream("C://SomeDir//notes.txt"))
        {
            byte[] buffer = text.getBytes();
            fos.write(buffer, 0, buffer.length);
        }
        catch(IOException ex){
            System.out.println(ex.getMessage());
        }
        System.out.println("The file has been written");
    }
}
```

- a. Запись строки в файл
- b. Чтение строки из файла
- c. Считывание в промежуточный буфер из массива символов
- d. Запись в промежуточный буфер из массива символов

Правильный ответ: a.

Задание 4

Выберите несколько правильных вариантов ответа.

Дан код. Выберите все правильные ответы, при которых код скомпилируется, если вставить их в строку 1: (Выберите 2 варианта ответа)

```
public class OverrideThrowsTest {
    public static void main(String[] args)
        // 1
    {
        A a = new A();
        a.method();
        A ab = new B();
        ab.method();
        B b = new B();
        b.method();
    }
}

class A {
    public void method() throws IOException {}
}

class B extends A {
    public void method() throws FileNotFoundException {}
}
```

- a. throws IOException
- b. throws Exception
- c. throw Exception
- d. throws FileNotFoundException

Правильный ответ: a, b

Задание 5

Выберите один правильный вариант ответа.

Дан код. Какой результат будет выведен на экран?

```
class A implements Cloneable{ //1
    public int i=10; }
class B extends A implements Cloneable{
    public int i=20;
    @Override
    public B clone() throws CloneNotSupportedException {
        B cloneA = (B) super.clone(); //2
        cloneA.i = 15;
        return cloneA; } }
public class MyClass {
    public static void main(String[] args) throws CloneNotSupportedException {
        B b = new B();
        A a = b.clone(); //3
        System.out.println(a.i); } }
```

- a. 15
- b. Ошибка компиляции в строке 2
- c. 10
- d. Ошибка выполнения в строке 2

Правильный ответ: c.

ОМ открытого типа

Задание 6

Дайте развернутый ответ.

Представьте реализацию алгоритма пузырьковой сортировки в Java-коде?

Правильный ответ:

```
3 public class Solution {
4     public static void main(String[] args) {
5         int[] testArr = new int[] {6,3,8,2,6,9,4,11,1};
6         bubbleSort( array: testArr);
7         for (int i : testArr) {
8             System.out.println( x:i );
9         }
10    }
11
12    public static void bubbleSort(int[] array) {
13        for(int i = array.length -1; i > 1; i--) {
14            for (int j = 0; j < i; j++) { //
15                if (array[j] > array[j+1]) {
16                    int temp = array[j];
17                    array[j] = array[j+1];
18                    array[j+1] = temp;
19                }
20            }
21        }
22    }
23 }
```

Задание 7

Дайте развернутый ответ.

Представьте реализацию алгоритма сортировка методом выбора на java.

Правильный ответ:

```
2 public class Solution {
3     public static void main(String[] args) {
4         int[] testArr = new int[]{6, 3, 8, 2, 6, 9, 4, 11, 1};
5         sortBySelect(array: testArr);
6         for (int i : testArr) {
7             System.out.println(x:i);
8         }
9     }
10    public static void sortBySelect(int[] array) {
11
12        for (int i = 0; i < array.length-1; i++) { // внешний обычный цикл
13            int min = i;
14            for (int j = i + 1; j < array.length; j++) { // обычный цикл, но с отчетом с сортированных
15                if (array[j] < array[min]) {
16                    min = j;
17                }
18            }
19            int temp = array[i]; // вставка отсортированного числа, в положеную ему ячейку
20            array[i] = array[min];
21            array[min] = temp;
22        }
23    }
24 }
```

Задание 8

Дайте развернутый ответ.

Что такое паттерны проектирования?

Правильный ответ:

Шаблон проектирования или паттерн (англ. design pattern) в разработке программного обеспечения — повторяемая архитектурная конструкция, представляющая собой решение проблемы проектирования в рамках некоторого часто возникающего контекста.

Задание 9

Дайте развернутый ответ.

Дайте определение понятию паттерн «фабричный метод»?

Правильный ответ:

Фабричный метод — это порождающий паттерн проектирования, который определяет общий интерфейс для создания объектов в суперклассе, позволяя подклассам изменять тип создаваемых объектов

Задание 10

Дайте развернутый ответ.

Дайте определение понятию шаблону проектирования синглтон?

Синглтон - это порождающий паттерн проектирования, который гарантирует, что у класса есть только один экземпляр, и предоставляет к нему глобальную точку доступа.

7.3. Оценочные средства для Промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр 4

№ 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.
31	Принципы ООП.

№ п/п	Вопросы к экзамену
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.
66	Работа с датой и временем в Java 7: класс Date.

№ п/п	Вопросы к экзамену
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, обход 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.

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
4	Экзамен (по накопительному рейтингу)	отлично	от 85 до 100 баллов
		хорошо	от 70 до 84 баллов
		удовлетворительно	от 55 до 69 баллов
		неудовлетворительно	от 0 до 54 баллов

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	Монахов, В. В.	Язык Программирования Java и среда NetBeans	Учебное пособие	2021	ЭБС "IPRbooks"
2	Мухаметзянов, Р. Р.	Основы Программирования на Java	Учебное пособие	2017	ЭБС "IPRbooks"
3	Вязовик, Н. А.	Программирование на Java	Учебное пособие	2021	ЭБС «IPRBooks»

8.2. Дополнительная 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	Гаряева В. В	Решение задач с использованием пакетов Прикладных Программ	Учебное пособие	2017	ЭБС "IPRbooks"
2	Лоскутов В. И., Коробова И.Л.	Разработка информационных систем для Windows Store		2016	ЭБС "IPRbooks"
3	Стешин А. И.	Информационные системы в организации	Учебное пособие	2019	ЭБС "IPRbooks"
4	Голицына О. Л., Максимов Н.В., Попов И.И.	Информационные системы	Учебное пособие	2018	ЭБС "Znaniy.com"

8.3. Перечень Профессиональных баз данных и информационных справочных систем

- About SWEBOOK. Режим доступа: <https://www.computer.org/web/swebok>,
- 2016-01-01.
- Java и вы. Режим доступа: <http://www.java.com/ru/>, 2016-01-01.
- Oracle Technology Network - Java. Режим доступа: <http://www.oracle.com/technetwork/java/index.html>, 2016-01-01.
- Project Management Institute. Режим доступа: <http://www.pmi.org/>, 2016-01-01.

8.4. Перечень Программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
1	Eclipse Foundation Eclipse версия 4	Лицензия Eclipse Public License
2	NetBeans Community NetBeans IDE версия 8	Лицензия LGPLv2.1, GPLv2 with Classpatch exception
3	The CodeBlocks team CodeBlocks версия 16-	Лицензия GNU GPLv3

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-408).	Компьютер (монитор 17", системный блок Intel (R) Celeron (R) 2,66 GHz / 1 Gb / 80 Gb), маршрутизатор 2801 Router, коммутатор Catalyst, экран/интерактивная доска Smart Board TV, Проектор Acer P1303W., стол Преподавательский, стол ученический, стол компьютерный, стул, доска аудиторная (маркерная).
2	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	Стол ученический двухместный (моноблок), стол Преподавательский, стул, доска аудиторная (меловая).

№ 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
	Промежуточной аттестации (УЛК-413).	
3	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).	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.
4	Помещение для самостоятельной работы Студентов (Г-401)	Стол ученический, стул, компьютер с выходом в сеть интернет.