

ПРИЛОЖЕНИЕ

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Рязанский государственный радиотехнический университет имени В.Ф. Уткина»**

КАФЕДРА «ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ»

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
«Промышленное программирование»**

Направление

09.03.01 Информатика и вычислительная техника

Направленность (профиль) подготовки
Вычислительные машины, комплексы, системы и сети

Квалификация (степень) выпускника — Бакалавр

Форма обучения — очная, заочная

Рязань

1. ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной профессиональной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям ОПОП.

Основная задача – обеспечить оценку уровня сформированности общекультурных, общепрофессиональных и профессиональных компетенций.

Контроль знаний обучающихся проводится в форме промежуточной аттестации.

Промежуточный контроль по дисциплине осуществляется проведением экзамена.

2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформированность каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

1) пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;

2) продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;

3) эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

Уровень освоения компетенций, формируемых дисциплиной:

а) описание критериев и шкалы оценивания тестирования:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 85 до 100%
2 балла (продвинутый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 75 до 84%
1 балл (пороговый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 60 до 74%
0 баллов	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 0 до 59%

б) описание критериев и шкалы оценивания теоретического вопроса:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя.
2 балла (продвинутый уровень)	выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов.
1 балл (пороговый уровень)	выставляется студенту, который дал неполный ответ на вопрос в билете и смог ответить на дополнительные вопросы только с помощью преподавателя.
0 баллов	выставляется студенту, который не смог ответить на вопрос

в) описание критериев и шкалы оценивания практического задания:

Шкала оценивания	Критерий
------------------	----------

3 балла (эталонный уровень)	Задача решена верно
2 балла (продвинутый уровень)	Задача решена верно, но имеются технические неточности в расчетах
1 балл (пороговый уровень)	Задача решена верно, с дополнительными наводящими вопросами преподавателя
0 баллов	Задача не решена

На экзамен выносятся: тестовое задание, 1 практическое задание и 1 теоретический вопрос. Студент может набрать максимум 9 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Шкала оценивания	Критерий	
отлично (эталонный уровень)	8 – 9 баллов	Обязательным условием является выполнение всех предусмотренных в течение семестра заданий
хорошо (продвинутый уровень)	6 – 7 баллов	
удовлетворительно (пороговый уровень)	4 – 5 баллов	
неудовлетворительно	0 – 3 баллов	Студент не выполнил всех предусмотренных в течение семестра текущих заданий

3 ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
Раздел 1. Введение в разработку промышленного программного	ПК-2	Экзамен
Раздел 2. Виртуальная машина Java	ПК-2 ПК-3	Экзамен
Раздел 3. Алгоритмические средства языка Java	ПК-2 ПК-3	Экзамен
Раздел 4. Средства объектного программирования языка Java	ПК-2	Экзамен
Раздел 5. Средства объектно-ориентированного программирования языка Java	ПК-2	Экзамен
Раздел 6. Модульность и обобщенное программирование на Java	ПК-2	Экзамен
Раздел 7. Стандартная библиотека Java	ПК-2	Экзамен
Раздел 8. Многопоточное программирование на Java. Параллелизм	ПК-3	Экзамен
Раздел 9. Модель безопасности Java. Программирование распределенных приложений	ПК-3	Экзамен

4 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1. Промежуточная аттестация в форме экзамена

Код компетенции	Результаты освоения ОПОП Содержание компетенций
ПК-2	Способен проектировать и разрабатывать программное обеспечение

ПК-2.1. Проектирует и разрабатывает программное обеспечение

Типовые тестовые вопросы

1. Что из перечисленного не является системой программирования (выбрать два):
Delphi
C++ Bulider
Visual Studio
Java
2. Традиционным методом организации информационных систем является:
клиент-клиент
клиент-сервер
сервер-сервер
размещение всей информации на одном компьютере
3. Для повышения эффективности разработки программного обеспечения применяют:
Delphi
C++
CASE –средства
Pascal
4. По масштабу ИС подразделяются на:
малые, большие
сложные, простые
объектно- ориентированные и прочие
одиночные, групповые, корпоративные
5. Microsoft.Net является:
платформой
языком программирования
системой управления базами данных
прикладной программой

Типовые вопросы открытого типа:

1. JDBC – технология, призванная осуществлять связь программы, написанной на языке Java с ... (**базами данных**)
2. Приложения, позволяющие обмениваться данными в сети Интернет, называют ... (**web-приложениями**)
3. ORM — технология программирования, которая связывает ... с концепциями объектно-ориентированных языков программирования. (**базы данных**)
4. Программные средства, такие как Git, Subversion, Mercurial, относят к системам ... (**контроля версий**)
5. Семиуровневая модель спецификаций, описывающих архитектуру сети, образованной разными устройствами называется моделью ... (**Open System Interconnection reference model, OSI**)

ПК-2.2. Применяет современные инструментальные средства при разработке программного обеспечения

Типовые тестовые вопросы

1. Средством визуальной разработки приложений является:
Visual Basic
Pascal
Delphi
язык программирования высокого уровня
2. Под CASE – средствами понимают
программные средства, поддерживающие процессы создания и сопровождения программного обеспечения
языки программирования высокого уровня
среды для разработки программного обеспечения
прикладные программы
3. Вызов метода, переопределенного в наследнике, по интерфейсу родительского класса это:
Наследование;
Инкапсуляция;
Полиморфизм;
Вызов виртуального метода.
4. Какое ключевое слово используется, чтобы показать, что с методом может работать не более чем один поток одновременно?
Public
Private
Sealed
Synchronized
5. Что позволяет реализовать множественное наследие в Java?
Статические методы
Интерфейсы
Абстрактные классы
Внутренние классы

Типовые вопросы открытого типа:

1. Базовая часть исполняющей системы Java, которая интерпретирует байт-код Java, скомпилированный из исходного текста Java-программы для конкретной операционной системы называется ... (**JVM, Java Virtual Machine**)
2. Минимальный набор для исполнения приложений, включающий JVM, но без средств разработки – это ... (**JRE, Java Runtime Environment**)
3. Полный набор для разработки и запуска приложений, состоящий из компилятора, утилит, исполнительной системы JRE, библиотек, документации называется ... (**JDK, Java Development Kit**)
4. Комплексы программных средств, используемый программистами для разработки программного обеспечения, к которым можно отнести IntelliJIDEA, Visual Studio, Eclipse, называются ... (**Интегрированные среды разработки, Integrated development environment — IDE**)
5. За освобождение памяти в куче от хранения объектов, которые не используются программой отвечает ... (**сборщик мусора, garbage collector, GC**)

Код компетенции	Результаты освоения ОПОП Содержание компетенций
ПК-3	Способен разрабатывать отдельные компоненты системного программного обеспечения

ПК-3.1. Определяет перечень необходимой для создания инструментальных средств технической документации

Типовые тестовые вопросы

1. Как называется генератор документации в HTML-формате из комментариев исходного кода на Java?
Javadoc
Lombok
Maven
Gradle
2. Какой аннотацией javadoc помечается информация об авторах класса или интерфейса?
@version
@author
@return
@param
3. Какой аннотацией javadoc помечается информация о версии класса или интерфейса?
@version
@author
@return
@param
4. Какой аннотацией javadoc помечается информация о входном параметре метода?
@version
@author
@return
@param
5. Какой аннотацией javadoc помечается информация о возвращаемом значении метода?
@version
@author
@return
@param

Типовые вопросы открытого типа:

1. Системы, предназначенные для организации коллективного написания кода называют ... **(системами контроля версий)**
2. Согласно большинству соглашений об оформлении программного кода, классы в языке Java должны именоваться ... **(с заглавной буквы)**
3. Команда branch в системе git отвечает за ... **(создание новой ветки)**
4. Команда clone в системе git отвечает за ... **(создание копии / клонирование удалённого репозитория)**
5. Согласно большинству соглашений об оформлении программного кода, константы в языке Java должны именоваться в ... регистре. **(верхнем)**

ПК-3.2. Разрабатывает исходные коды создаваемых инструментальных средств программирования

Типовые тестовые вопросы

1. Дан код:

```
interface Readable {  
    void read();  
}  
abstract class Editor {  
    void edit(){}  
}  
interface Lexicon /* 1 */ /* 2 */ {  
    /* 3 */  
}
```

Какой набор строк, вставленный соответственно вместо комментариев 1, 2 и 3, позволит компилироваться интерфейсу Lexicon без ошибок? (выбрать один)

```
extends; Editor; public void edit(){}  
implements; Readable; void read(){}  
extends; Editor; protected void edit(){}  
implements; Readable; public void read(){} 
```

2. Даны объявления интерфейсов и класс, их реализующий:

```
interface First {  
    int i = 1;  
}  
interface Second {  
    int i = 2;  
}  
public classClazz implements Second, First {  
    public static void main(String[] args) {  
        System.out.println(i);  
    }  
}
```

Каким будет результат компиляции и выполнения? (выбрать один)

1
2

compilation fails
runtime error

3. Даны объявления класса и интерфейса:

```
interface Readable{  
    void read();  
}  
class Read {  
    public void read(){}  
}
```

Какие объявления классов и интерфейсов корректны? (выбрать два)

**interface Editable extends Readable{}
interface Editable implements Readable{}
class Edit implements Readable extends Read {}
class Edit extends Read implements Readable {}
class Edit implements Readable, Read {}
class Edit extends Readable, Read {}**

4. Даны объявления классов и интерфейсов:

```
interface First {}  
interface Second extends First {}  
class Klass implements Second{}  
classClazz implements First {}
```

Какие объявления ссылок корректны? (выбрать три)

**First a = (First)newClazz();
Second b = (First)newKlass();
First c = (First)newKlass();
Second d = (Second) newClazz();
Klass e = (Second)newClazz();**

5. Дан код:

```
public class A {  
    public static void main(String[] args){  
        String str = "Hello";  
        System.out.print(str);  
        doWork(str);  
        System.out.print(str);  
    }  
    public static void doWork(String value){  
        value = null;  
    }  
}
```

Что будет результатом компиляции и запуска? (выбрать один)

Hello
Hellonull

HelloHello

runtime error

Типовые вопросы открытого типа:

1. Отношение между классами, при котором характеристики одного класса (суперкласса) передаются другому классу (подклассу) без необходимости их повторного определения – это ... **(наследование)**
2. Класс может реализовать любое число интерфейсов, используя ключевое слово ... **(implements)**
3. Способность ссылки динамически определять версию переопределенного метода в зависимости от переданного по ссылке типа объекта называется ... **(полиморфизмом)**
4. Запрещено переопределять метод в порожденном классе, если в суперклассе он объявлен со спецификатором ... **(final)**
5. Для обращения к конструктору суперкласса и для доступа к полю или методу суперкласса применяется ключевое слово ... **(super)**

ПК-3.3. Выполняет сопровождение программного обеспечения инструментальных средств

Типовые тестовые вопросы

1. Дан код:

```
String s = new String("3");  
System.out.println(1 + 2 + s + 4 + 5);
```

В результате при компиляции и запуске будет выведено (выбрать один):

12345
3345
1239
339
15
compilation fails

2. Дан фрагмент кода:

```
String s1 = new String("Java");  
String s2 = "Java";СТРОКИ  
String s3 = new String(s1);  
String s4 = "Java";
```

Какие из предложенных операторов дадут результат true? (выбрать два)

s1 == s2
s1 == s3
s2 == s4
s2 == s3
s2.equals(s1)

3. Дан код:

```
String[] strings = new String[]{"a", "b", "c"};  
int k = 0;  
for (String element : strings) {  
    strings[k].concat(String.valueOf(k));  
    ++k;  
}  
System.out.print(Arrays.toString(strings));
```

Что будет выведено на консоль при компиляции и выполнении кода? (выбрать один)

[a, b, c]
[a0, b1, c2]
[a1, b2, c3]
compilation fails

4. Дан фрагмент кода:

```
String st = "0";  
StringBuffer sb = new StringBuffer("a");  
// 1  
// 2  
System.out.print(st);  
System.out.print(sb);
```

Какой из фрагментов кода, будучи вставленным вместо комментария 1 и 2 соответственно, выведет в консоль 01ab? (выбрать один)

```
st = st.concat("1");sb.append("b");  
st.concat("1");  
sb.append("b");  
st = st.concat("1");  
sb.concat("b");  
st = st.append("1");  
sb.append("b");
```

5. Дан фрагмент кода:

```
long result = Arrays.stream(new String[]{"JSE", "JDK", "J"}) // line 1  
.filter(s -> s.length() > 1)  
.filter(s -> s.contains("J"))  
.count();
```

Какое значение примет result? (выбрать один)

0
2
3
compilation fails at line 1

Типовые вопросы открытого типа:

1. Методология программирования, основанная на функционировании программного продукта как результата взаимодействия совокупности объектов, каждый из которых является экземпляром конкретного класса называется ... (**объектно-ориентированное программирование, ООП**)
2. Именованная модель реальной сущности, обладающая конкретными значениями свойств и проявляющая свое поведение – это ... (**объект**)
3. Модель информационной сущности, представляющая универсальный тип данных, состоящая из набора полей данных и методов их обработки – это (**класс**)
4. Виртуальная машина Java содержит две основных области памяти - ... и ... (**стэк и куча**)
5. Выполнение любой программы в языке Java начинается с функции с именем ... (**main()**)

Типовые теоретические вопросы для экзамена по дисциплине

- 1) Виртуальная машина Java. Понятие виртуальной машины.
- 2) Виртуальная машина Java. Среда исполнения и байт-код.
- 3) Виртуальная машина Java. Взаимодействие виртуальной машины с операционной системой.
- 4) Виртуальная машина Java. Структура приложений на Java.
- 5) Алгоритмические средства языка Java. Базовые типы языка.
- 6) Алгоритмические средства языка Java. Преобразования типов в выражениях.
- 7) Алгоритмические средства языка Java. Особенности инициализации массивов.
- 8) Алгоритмические средства языка Java. Операторы управления памятью. Жизненный цикл и этапы разработки программного обеспечения.
- 9) Разработка технического задания.
- 10) Средства ООП языка Java. Реализация интерфейсов как альтернатива множественному наследованию.
- 11) Средства ООП языка Java. Основные принципы и типы исключительных ситуаций.