

Вопрос 1

Что будет напечатано при компиляции и выполнении данного кода?

```
String d = "beekeeper";  
d.substring(1,7);  
d = "w" + d;  
d.insert(3, "bee");  
System.out.println(d);
```

- А) weebeekeep
- Б) wbeekeeper
- В) wbebeeekeeper
- Г) weekeep
- Д) Ошибка компиляции**
- Е) Возникнет исключение

Вопрос 2

Какой будет результат выполнения кода:

```
public class Main {  
    public static void main(String[] args) {  
        double d = Math.sqrt(-1);  
        System.out.println(Double.NaN == d);  
        d = d / 0;  
        System.out.println(Double.isNaN(d));  
    }  
}
```

A) true
false

Б) true
true

В) false
false

Г) false
true

Вопрос 3

Что произойдет при компиляции программы (может быть несколько ответов):

```
import static com.sun.PkgAccess.*;

public class PkgAccess2 {
    int x1 = PkgAccess.tiger; //1
    int x2 = tiger; //2
    int x3 = com.sun.PkgAccess.tiger; //3
    int x4 = sun.PkgAccess.tiger; //4
}
```

Если следующий класс находится в другом пакете:

```
package com.sun;
public class PkgAccess {
    public static int tiger = 1414;
}
```

А) Программа скомпилируется

Б) Ошибка компиляции в строке 1

В) Ошибка компиляции в строке 2

Г) Ошибка компиляции в строке 3

Д) Ошибка компиляции в строке 4

Вопрос 4

Что напечатает следующий фрагмент кода:

```
int x = 0;  
int y = 10;  
do {  
    y--;  
    ++x;  
} while (x < 5);  
System.out.print(x + "," + y);
```

A) 6,5

Б) 5,5

В) 6,6

Г) 5,6

Вопрос 5

Какие из нижеследующих сигнатур могут быть использованы для объявления метода `main()`, который будет точкой входа при запуске приложения? (Выберите все допустимые варианты)

1. `public static void main()`
- 2. `public static void main(String arg[])`**
3. `public void main(String [] arg)`
- 4. `public static void main(String[] args)`**
5. `public static int main(String [] arg)`

Вопрос 6

Какой результат будет получен при компиляции и выполнении данного кода (может быть несколько ответов):

```
public class Main {  
    public static void main(String [] str) {  
        Boolean b1 = new Boolean("false");           //1  
        if (b1 = true) System.out.println("b1=" + b1); //2  
  
        boolean b2 = false;  
        if (b2 = true) System.out.println("b2=" + b2); //3  
  
        int i = 1;  
        if (i = 2) System.out.println("i=" + i);      //4  
    }  
}
```

1. Ошибка времени выполнения
2. Ошибка компиляции в строке 1
3. Ошибка компиляции в строке 2
4. Ошибка компиляции в строке 3
- 5. Ошибка компиляции в строке 4**

Вопрос 7

Какой результат выполнения программы:

```
public class StartClass {  
    public static void main(String[] args) {  
        Double d1 = 1d;  
        Double d2 = 1d;  
        System.out.println(d1 == d2);  
    }  
}
```

1. true

2. false

Вопрос 8

Что будет выведено на экран, если
скомпилировать и запустить следующий код:

```
public class Main {  
    public static void main(String s[]) {  
        int anar[] = new int[5];  
        System.out.println(anar[0]);  
    }  
}
```

1. 5
- 2. 0**
3. Null
4. Возникнет ошибка компиляции

Вопрос 9

Каким будет результат компиляции и запуска следующей программы?

```
public class App1 {  
    public static void main(String[] args) {  
        System.out.println(addToString("12345678910",','));  
    }  
    public static StringBuffer addToString(String s, char c) {  
        StringBuffer b = new StringBuffer(s);  
        int p = 0;  
        for (int i = 1; i < b.length(); i++) {  
            if (i%3 == 0) {  
                b.insert(b.length()-i-p, c);  
                p++;  
            }  
        }  
        return b;  
    }  
}
```

1. Компиляция пройдет успешно и на экране напечатается "12,345,678,910" без кавычек
2. Компиляция пройдет успешно и на экране напечатается 123,456,789,10 без кавычек
3. Ошибка компиляции
- 4. Компиляция пройдет успешно, но во время выполнения возникнет `ArrayIndexOutOfBoundsException`**
5. Компиляция пройдет успешно, но во время выполнения возникнет `StackOverflowError`

Вопрос 10

В каком порядке должны размещаться следующие операторы?

1. **1.Package statement**
 2. **Imports**
 3. **Class or interface definitions**
2. 1. Imports
 2. Package statement
 3. Class or interface definitions
3. любой порядок

Вопрос 11

Что будет выведено следующим кодом?

```
public class Test1 {  
    public static void main(String args[]) {  
        System.out.println( 5 ^ 6 & 3);  
    }  
}
```

1. 2
2. 3
3. 4
4. 5
5. 6
- 6. 7**
7. 8

Вопрос 12

В какой строке(-ах) кода содержится ошибка:

```
int i = 1;          //1
```

```
i = -+(10 + 2 + i); //2
```

```
++i--;             //3
```

```
System.out.println(i);
```

1. 1

2. 2

3. 3

4. 2,3

5. Код не содержит ошибок

Вопрос 13

Что будет выведено на экран в результате выполнения программы ?

```
public class Main {  
    public static void main(String[] args) {  
        try {  
            String testString = "This is a test string.";  
            String[] resultArray = testString.split(" ");  
            System.out.println(resultArray.length);  
        }  
        catch (Exception ex) {  
            System.out.println("exception");  
        }  
    }  
}
```

1. Ошибка компиляции
2. exception
3. 4
- 4. 5**

Вопрос 14

Что произойдёт если скомпилировать следующий код?

```
public class MyFirst {  
    public static void main(String[] args){  
        int i, j;  
        i = 10;  
        j = 20;  
        while(++i < --j);  
        System.out.println(i);  
    }  
}
```

1. Вывод на экран 10
- 2. Вывод на экран 15**
3. Вывод на экран 20
4. Ошибка компиляции, так как отсутствует тело цикла.

Вопрос 15

Что будет выведено на консоль?

```
public class Main {  
    public static void main(String[] args) {  
        int a = 10;  
        int b = 100;  
        double c = (double) (a/b);  
        String str = String.format("%.4f", c);  
        System.out.println(str);  
    }  
}
```

1. 0,1
- 2. 0,0000**
3. 0,1000
4. 0

Вопрос 16

Гарантированно ли начнется сборка мусора после вызова `Runtime.getRuntime().gc()`?

1. Да
- 2. Нет**

Вопрос 17

Каким будет результат выполнения следующей программы?

```
import java.util.*;
public class TestFormatter {
    public static void main(String... args){
        Integer l1 = 0;
        Integer l2 = -1;
        Integer l3 = 1;
        Formatter f = new Formatter();
        f.format("%1$b ", l1.toString())
        .format("%1$b ", l2.toString())
        .format("%1$b ", l3.toString());
        System.out.println(f.toString());
    }
}
```

1. false true true
2. true false true
3. 0 -1 1
4. false false true
- 5. true true true**

Вопрос 18

Какой будет результат компиляции и выполнения следующего кода?

```
public class Test {  
    public static void main(String[] args) {  
        for (final int i : new int[] { 1, 2, 3 }) {  
            System.out.println(i + 1);  
        }  
    }  
}
```

1. **2**
3
4
2. Ошибка компиляции
3. 1
2
3
4. Ошибка времени выполнения

Вопрос 19

Выведется ли текст, помещенный в блок условного оператора?

```
public static void main(String[] args){  
    int x = 18;  
    int y = x++;  
    if (x == 18 && y > 10){  
        System.out.println("y = " + y);  
        System.out.println("x = " + x);  
    }  
}
```

1. нет

2. да

Вопрос 20

В какой строке возникнет ошибка компиляции:

```
public class Test {  
    public static void main(String[] args) { // 2  
        char a = '\u005B'; // 3 (Скобка [ )  
        char c = '\u005D'; // 4 (Скобка ] )  
        System.out.println(a * c); // 5  
    }  
}
```

1. 2

2. 3

3. 4

4. 5

5. Код скомпилируется без ошибок.

Вопрос 21

Что выведет следующий код?

```
public class Main {  
    public static void main(String[] args) {  
        recur(99);  
    }  
    public static void recur(int a) {  
        if (a <= 100) {  
            System.out.print("a=" + a);  
            recur(++a);  
            System.out.print("a=" + a);  
        }  
    }  
}
```

1. a=99 a=100 a=100 a=100
- 2. a=99 a=100 a=101 a=100**
3. a=99 a=100 a=100 a=99
4. a=99 a=100 a=100 a=101

Вопрос 22

Как определить длину массива myarray?

1. myarray.size()
- 2. myarray.length**
3. myarray.size
4. myarray.length()