

6. Все трехзначные числа, в десятичной записи которых нет одинаковых цифр.
7. Наибольший общий делитель и наименьшее общее кратное этих чисел.
8. Простые числа.
9. Отсортированные числа в порядке возрастания и убывания.
10. Числа в порядке убывания частоты встречаемости чисел.
11. “Счастливые” числа.
12. Числа Фибоначчи: $f_0 = f_1 = 1, f(n) = f(n-1) + f(n-2)$.
13. Числа-палиндромы, значения которых в прямом и обратном порядке совпадают.
14. Элементы, которые равны полусумме соседних элементов.
15. Период десятичной дроби $p = m/n$ для первых двух целых положительных чисел n и m , расположенных подряд.
16. Построить треугольник Паскаля для первого положительного числа.

Тестовые задания к главе 1

Вопрос 1.1.

Дан код:

```
class Quest1 {
    private static void main (String a) {
        System.out.println("Java 2");
    }
}
```

Какие исправления необходимо сделать, чтобы класс **Quest1** стал запускаемым приложением? (выберите 2 правильных варианта)

- 1) объявить класс **Quest1** как **public**;
- 2) заменить параметр метода **main()** на **String[] a**;
- 3) заменить модификатор доступа к методу **main()** на **public**;
- 4) убрать параметр из объявления метода **main()**.

Вопрос 1.2.

Выберите истинные утверждения о возможностях языка Java: (выберите 2)

- 1) возможно использование оператора **goto**;
- 2) возможно создание метода, не принадлежащего ни одному классу;
- 3) поддерживается множественное наследование классов;
- 4) запрещена перегрузка операторов;
- 5) поддерживается многопоточность.

Вопрос 1.3.

Дан код:

```
class Quest3 {
    public static void main(String s[ ]) {
        String args;
        System.out.print(args + s);
    }
}
```

Результатом компиляции кода будет:

- 1) ошибка компиляции: метод **main()** содержит неправильное имя параметра;

- 2) ошибка компиляции: переменная **args** используется до инициализации;
- 3) ошибка компиляции: несовпадение типов параметров при вызове метода **print()**;
- 4) компиляция без ошибок.

Вопрос 1.4.

Дан код:

```
public class Quest4 {  
    public static void main(String[] args) {  
        byte b[]=new byte[80];  
        for (int i=0;i<b.length;i++)  
            b[i]=(byte)System.in.read();  
        System.out.print("Ok");  
    }  
}
```

Результатом компиляции и запуска будет:

- 1) вывод: Ok;
- 2) ошибка компиляции, так как метод **read()** может порождать исключительную ситуацию типа **IOException**;
- 3) ошибка компиляции, так как длина массива **b** может не совпадать с длиной считываемых данных;
- 4) ошибка времени выполнения, так как массив уже проинициализирован.

Вопрос 1.5.

Дан код:

```
public class Quest5{  
    public static void main(){  
        System.out.print("A"); }  
    public static void main(String args){  
        System.out.print("B"); }  
    public static void main(String[] args){  
        System.out.print("B"); } } }
```

Что будет выведено в результате компиляции и запуска:

- 1) ошибка компиляции;
- 2) Б;
- 3) БАА;
- 4) В;
- 5) АВВ.