Išplėstiniai Java klausimai

Išplėstinis klausimas #1

```
interface ? {
    boolean hasNext();
    Object next();
}
```

Išplėstinis klausimas #2

```
public class Main {

   public static void main(String[] args) {
       B b = new B();
   }

   static class A {
       public A() {
           System.out.println("A");
       }
   }

   static class B extends A {
       public B() {
           System.out.println("B");
       }
   }
}
```

Išplėstinis klausimas #3

```
class A {}
class B extends A {}
interface I {}
class C extends B implements I {}
class D extends C {}
```

Išplėstinis klausimas #4

```
public class Main {
```

```
    public static void main(String[] args) {
        int i = 010;
        int j = 07;
        System.out.println(i + j);
    }
}
```

Išplėstinis klausimas #5

```
public class Main {

    public static void main(String[] args) {
        A a = new B();
        a.print();
    }

    static class A {
        public void print() {
            System.out.println("A");
        }
    }

    static class B extends A {
        public void print() {
            System.out.println("B");
        }
    }
}
```

Išplėstinis klausimas #6

```
public class Main {
    public static void main(String[] args) {
        Printer p = new Printer();
        p.print(new B(), new C());
    }

    static class A {}
    static class B extends A {}
    static class C extends B {}

    static class Printer {
        public void print(A a, B b) {
```

```
        System.out.println("AB");
    }

    public void print(C c, A a) {
        System.out.println("BC");
    }

    public void print(A a, C c) {
        System.out.println("AC");
    }
  }
}
```

Išplėstinis klausimas #7

```
public class Main {
  public static void main(String[] args) {
    int array[] = new int[]{10, 11, 88, 2, 12, 120};
    int val = getSearch(array);
    System.out.println("Value is: " + val);
  }

  private static int getSearch(int[] inputArray) {
    int searchValue = inputArray[0];
    for (int i = 1; i < inputArray.length; i++) {
      if (inputArray[i] > searchValue) {
        searchValue = inputArray[i];
      }
    }
    return searchValue;
  }
}
```

Išplėstinis klausimas #8

```
Stream.of(1, 2, 3, 4, 5)
    .filter(e -> e % 3 == 1)
    .forEach(System.out::println);
```

Išplėstinis klausimas #9

```
Stream.of(10, 20, 30)
```

```
        .map(e -> e*e)
        .forEach(System.out::println);
```

Išplėstinis klausimas #10

```
public class Main {
    public static void main(String[] args) {
        int a = 2;
        int b = 4;
        a = a ^ b;
        b = a ^ b;
        a = a ^ b;
        System.out.println(a + “, “ + b);
    }
}
```

Išplėstinis klausimas #11

```
public class Main {
    public static void main(String[] args) {
        Integer[] arr = {6, 4, 5, 2};
        Set<Integer> set = new TreeSet<Integer>(Arrays.asList(arr));
        set.add(8);
        for (Integer value : set) {
            System.out.print(value + ", ");
        }
    }
}
```