

Paradygmaty programowania - ćwiczenia

Lista 6

1. (Scala) Zapisz w języku Scala zaprezentowane na wykładzie funkcje swap, partition, quick i quicksort, zachowując ich styl programowania. Funkcje nie muszą być polimorficzne.
2. (Scala) Jedną z pętli w języku Scala ma następującą składnię: *while (warunek) wyrażenie*. Napisz w Scali funkcję whileLoop, która pobiera dwa argumenty: warunek i wyrażenie i symuluje działanie pętli while. Jakiego typu muszą być argumenty i wynik funkcji?

Działanie programów z zadań 3-4 należy wyjaśniać, rysując „obrazy pamięci” tych programów, tzn. rysując referencje w postaci strzałek, komórki pamięci i ich zawartości.

3. Co i dlaczego wydrukuje poniższy program w Javie?

```
public class Porównanie{
    public static void main(String[] args){
        String s1 = "foo";
        String s2 = "foo";
        System.out.println(s1 == s2);
        System.out.println(s1.equals(s2));
        String s3 = new String("foo");
        System.out.println(s1 == s3);
        System.out.println(s1.equals(s3));
    }
}
```

4. Co i dlaczego wydrukuje poniższy program w Javie?

```
public class Aliasy{
    public static void main(String[] args){
        int[] ints = {1,2,3};
        for(int i : ints) {
            System.out.println(i); i = 0;
        }
        for(int i : ints)
            System.out.println(i);
        int[] ints2 = ints;
        for(int i=0; i<ints2.length; i++) {
            System.out.println(ints2[i]); ints2[i] = -1;
        }
        for(int i : ints)
            System.out.println(i);
    }
}
```