

## Ćwiczenia z programowania obiektowego

16.03.2011 r.

### Część 1. Tablice

1. Który z poniższych fragmentów kodu deklaruje tablicę liczb całkowitych o nazwie `img`?

- (a) `int img;`
- (b) `int[] img;`
- (c) `new int img[];`
- (d) `int img = int[];`

2. Czy w czasie pisania programu programista musi wiedzieć, jakiej wielkości będzie tablica?

3. Dana jest następująca tablica:

```
byte[] data = { 12, 34, 9, 0, -62, 88 };
```

Ile wynosi wartość `data.length`?

4. Jaki będzie wynik działania następującego fragmentu kodu?

```
int[] ar = {2, 4, 6, 8 };  
System.out.println( ar[0] + " " + ar[1] );
```

5. Jaki będzie wynik działania następującego fragmentu kodu?

```
int[] y = new int[10];  
  
y[0] = 34;  
y[1] = 88;  
  
System.out.println( y[0] + " " + y[1] + " " + y[5] );
```

6. Uzupełnij luki w podanym fragmencie kodu tak, aby elementy tablicy zostały wypisane w kolejności od ostatniego do pierwszego elementu.

```
int[] egArray = { 2, 4, 6, 8, 10, 1, 3, 5, 7, 9 };  
  
for ( int index= _____ ; _____ ; _____ )  
    System.out.print( egArray[ index ] + " " );
```

7. Dana jest tablica.

```
double[][] values =  
    { {1.2, 9.0, 3.2},  
      {9.2, 0.5, 1.5, -1.2},  
      {7.3, 7.9, 4.8} } ;
```

- (a) Jaką wartość ma `values[2][1]`?
- (b) Jaką wartość ma `values[3][0]`?
- (c) Ile wynosi `values.length`?
- (d) Ile wynosi `values[2].length` ?

Która z poniższych instrukcji tworzy tablicę o 5 wierszach długości 7 kolumn każda.

- (a) `stuff = new stuff[5][7] ;`
- (b) `stuff = new long[5][7] ;`
- (c) `stuff = long[5][7] ;`
- (d) `stuff = long[7][5] ;`

8. Które z poniższych wyrażeń tworzy i przypisuje na zmienną `array` dwuwymiarową tablicę o 7 wierszach, ale nie tworzy jeszcze wierszy?

- (a) `int[][] array = new int[7][];`
- (b) `int[][] array = new int[7];`
- (c) `int[][] array = new int[][7];`
- (d) `int[] array[7] = new int[];`

9. Napisz (na dwa różne sposoby) kod, który utworzy następującą tablicę.

```

12      -9      8
7       14
-32     -1      0

```

10. Dana jest deklaracja.

```
long[][] stuff ;
```

11. Dana jest następująca tablica.

```
int[][] items =
{ {0, 1, 3, 4},
  {4, 3, 99, 0, 7 },
  {3, 2} } ;
```

Które z poniższych wyrażeń zastępuje w tablicy `items` wiersz 0 przez zupełnie nowy wiersz?

- (a) `items[0][0] = 8; items[0][1] = 12; items[0][2] = 6;`
- (b) `items[0] = { 8, 12, 6 };`
- (c) `items[0] = new { 8, 12, 6 };`
- (d) `int[] temp = { 8, 12, 6 };
items[0] = temp;`

## Część 2. Obiekty

1. Jak zmienia się diagram obiektów podczas wykonania następującego kodu?

```
01: public class Pair {
02:     private int x;
03:     private int y;
04:
05:     public Pair(int x, int y) {
06:         this.x = x;
07:         this.y = y;
08:     }
09: }
10:
11: public class Main {
12:
13:     public static void main(String args[]) {
14:         Pair a;
15:         Pair b;
16:
17:         a = new Pair(1,2);
18:         b = new Pair(3,4);
19:
20:         Pair[] pairs;
21:
22:         pairs = new Pair[10];
23:         pairs[0] = new Pair(1,4);
24:         pairs[1] = a;
25:
26:         a = b;
27:
28:         pairs[1].x = 7;
29:         pairs[1] = new Pair(2,3);
30:         pairs[1] = null;
31:     }
32: }
```

- (a) Jak wygląda tablica `pairs` w linii 21?
- (b) Jak wygląda tablica `pairs` w linii 22?
- (c) Ile wynosi `pairs[7]` w linii 23? A `pairs[10]`?

2. Jaki jest wynik wykonania następującego kodu?

```
public class Mixer {
    public Mixer() { }
    public Mixer(Mixer m) { m1 = m; }
    Mixer m1;

    void go() { System.out.print("hi "); }

    public static void main(String[] args) {
        Mixer m2 = new Mixer();
        Mixer m3 = new Mixer(m2); m3.go();
        Mixer m4 = m3.m1;
        Mixer m5 = m2.m1;
        m5.go();
    }
}
```

### Część 3. Przekazywanie parametrów w Javie

1. Jaki jest wynik działania następującego programu?

```
01: class Cup {
02:     String type;
03:     Cup(String type) {
04:         this.type = type;
05:     }
06: }
07:
08: public class Test {
09:     static void fillWithCoffee(Cup cup) {
10:         cup.type = "coffee";
11:     }
12:
13:     static void makeSure(int certainty) {
14:         certainty = 100;
15:     }
16:
17:     public static void main(String[] args) {
18:         Cup cup = new Cup("tea");
19:         int certainty = 50;
20:
21:         fillWithCoffee(cup);
22:         makeSure(certainty);
23:
24:         System.out.println("I'm " + certainty + "% certain" +
25:             " the cup is filled with " + cup.type + ".");
26:     }
27: }
```

2. Dany jest następujący kod. Na końcu metody `test()`, zmienna `car` przechowuje referencję do jednego z obiektów klasy `Car`. Do którego?

```
01: void test() {
02:     Car car = new Car("Red");
03:     helper(car);
04: }
05:
06: void helper(Car c) {
07:     c = new Car("Black");
08:     anotherHelper(c);
09: }
10:
11: void anotherHelper(Car c) {
12:     c = new Car("White");
13: }
```

3. Co zostanie wypisane w linii 10?

```
01: public class FollowMe{
02:     int a;
03:     int b;
04:
05:     public void funct(){
06:         a=1;
07:         b=1;
08:         int[] c={2};
09:         changeMe(b,c);
10:         System.out.println(a+" "+b+" "+c[0]+" ");
11:     }
12:
13:     public void changeMe(int x,int[] c){
14:         a=2;
15:         x=2;
16:         c[0]=1;
17:     }
18:
19:     public static void main(String [] args){
20:         FollowMe obj=new FollowMe();
21:         obj.funct();
22:     }
23: }
```

4. Jaki jest wynik wykonania następującego kodu?

```
class Fizz {
    int x = 5;
    public static void main(String[] args) {
        Fizz f1 = new Fizz();
        Fizz f2 = new Fizz();
        Fizz f3 = fizzSwitch(f1,f2);
        System.out.println((f1 == f3) + " " + (f1.x == f3.x) +
            " " + f1.x);
    }
    static Fizz fizzSwitch(Fizz x, Fizz y) {
        Fizz z = x;
        z.x = 6;
        return z;
    }
}
```