

Materiał pomocniczy do kursu „Podstawy programowania”

Autor: Grzegorz Góralski

ggoralski.pl

Tablice cz. II

Kopiowanie tablic, tablice wielowymiarowe

Kopiowanie tablic

* Wykonaj kod:

```
int[] tablica1 = new int[10];  
int[] tablica2 = tablica1;  
  
for (int i=0; i<tablica1.length; i++) tablica1[i]=i;  
  
System.out.print("Tablica1: ");  
for (int i : tablica1) System.out.print(i+" ");  
System.out.println();  
System.out.print("Tablica2: ");  
for (int i : tablica2) System.out.print(i+" ");
```

* Wynik:

```
Tablica1: 0 1 2 3 4 5 6 7 8 9  
Tablica2: 0 1 2 3 4 5 6 7 8 9
```

Zauważ, że **najpierw** skopiowaliśmy tablicę, a **później** wypełniliśmy tablicę `tablica1` wartościami od 0 do 9

Tak naprawdę nie skopiowaliśmy wartości z jednej tablicy do drugiej, ale utworzyliśmy dwie **referencje** które wskazują na **tą samą** tablicę

Kopiowanie tablic

- * Dodaj i wykonaj kod:

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

```
tablica1[2] = 100;
```

```
System.out.print("Tablica1: ");
```

```
for (int i : tablica1) System.out.print(i+" ");
```

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

```
System.out.print("Tablica2: ");
```

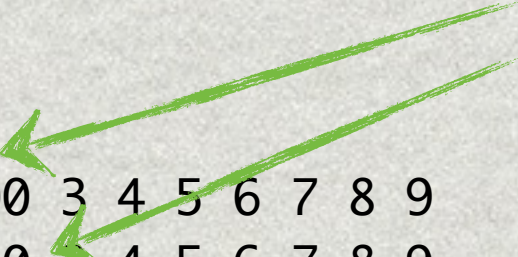
```
for (int i : tablica2) System.out.print(i+" ");
```

- * Wynik:

```
Tablica1: 0 1 100 3 4 5 6 7 8 9
```

```
Tablica2: 0 1 100 5 4 5 6 7 8 9
```

Jak widać, kiedy zmieniliśmy wartość komórki w jednej tablicy, zmieniła się także wartość komórki w drugiej tablicy



Kopiowanie tablic

- * Jak więc skopiować tablice? Można tak:

```
int[] tablica3 = new int[10];  
int[] tablica4 = new int[10];
```

```
for (int i=0; i<tablica3.length; i++) tablica3[i]=i;
```

```
for (int i=0; i<tablica3.length; i++) tablica4[i]=tablica3[i];
```

```
tablica3[2]=100;
```

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

```
System.out.print("Tablica3: ");
```


```
for (int i : tablica3) System.out.print(i+" ");
```

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

```
System.out.print("Tablica4: ");
```

```
for (int i : tablica4) System.out.print(i+" ");
```

Iterujemy obie tablice, pobieramy wartości z jednej i umieszczamy je w drugiej.



Sprawdzamy...




- * Wynik:

```
Tablica3: 0 1 100 3 4 5 6 7 8 9
```

```
Tablica4: 0 1 2 3 4 5 6 7 8 9
```

I o to właśnie chodzi!



Kopiowanie tablic - inne sposoby

- * Kopiowanie za pomocą `Arrays.copyOf()`;

```
import java.util.Arrays;
```

```
...
```

```
int[] tablicaKopia = Arrays.copyOf(tablica, tablica.length);
```


- * Kopiowanie za pomocą `System.arraycopy()`;

```
int [] tablicaKopia = new int[tablica.length];
```

Od którego elementu

Ile elementów

```
System.arraycopy( tablica, 0, tablicaKopia, 0, tablica.length );
```



- * Kopiowanie za pomocą `.clone()`;

```
int[] tablicaKopia = (int []) tablica.clone;
```

Zadanie

- * Napisz program który:
 - * Utworzy tablicę z liczbami typu double;
 - * Utworzy drugą tablicę, w której umieści wartości z tablicy pierwszej pomniejszone o średnią wszystkich liczb z tablicy pierwszej.
 - * Wypisze obliczone wartości.

Zadanie

* Plan:

1. Utworzyć tablicę z liczbami typu double (tablica1);
2. Utworzyć tablicę o takiej samej wielkości (tablica2).
3. Obliczyć sumę liczb z tablicy 1:
 - ◆ iterować tablicę1, dodając do zmiennej (suma) poszczególne wartości
4. Obliczyć średnią: $\text{suma} / \text{długość_tablicy1}$
 - ◆ $\text{suma} / \text{tablica1.length}$
5. Wypełnić tablicę2 wartościami:
 - ◆ iterujemy po obu tablicach w jednej pętli for():
 - pobieramy wartość z komórek z tablicy1
 - do tablicy2 wpisujemy do komórki o tym samym indeksie różnicę wartości z tablicy1 i obliczonej średniej
6. Wypisujemy wartości: suma, srednia i wartosci z tabicy2

Zadanie

```
double[] tablica1 = {1.0,3.5,3.4,4.9,8.2,1.0,4.1,8.7,9.2,10.1};
double[] tablica2 = new double[tablica1.length];
double suma = 0.0;
double srednia = 0.0;
// obliczamy sumę
for (int i = 0; i < tablica1.length; i++) {
    suma=suma+tablica1[i];
}
// obliczamy średnią
srednia = suma/tablica1.length;
// wypełniamy tablicę2 różnicą wartości z tablicy1 i średniej
for (int i = 0; i < tablica1.length; i++) {
    tablica2[i] = tablica1[i]-srednia;
}
// Wypisujemy rezultaty
System.out.println("Suma: "+suma);
System.out.println("Srednia: "+srednia);
for (int i = 0; i < tablica2.length; i++) {
    System.out.print(tablica2[i]+" ");
}
}
```


Tablice wielowymiarowe

- * Wypełnienie tablicy liczbami od 0 do 99:

```
int[][] tablica = new int[10][10];  
  
int wartosc = 0;  
for (int i=0; i<10; i++) {  
    for (int j=0; j<10; j++ ) {  
        tablica[i][j] = wartosc;  
        wartosc++;  
    }  
}
```

	0	1	2	3	4	5	6	7	8	9
0	0	1	2	3	4	5	6	7	8	9
1	10	11	12	13	14	15	16	17	18	19
2	20	21	22	23	24	25	26	27	28	29
3	30	31	32	33	34	35	36	37	38	39
4	40	41	42	43	44	45	46	47	48	49
5	50	51	52	53	54	55	56	57	58	59
6	60	61	62	63	64	65	66	67	68	69
7	70	71	72	73	74	75	76	77	78	79
8	80	81	82	83	84	85	86	87	88	89
9	90	91	92	93	94	95	96	97	98	99

Tablice wielowymiarowe

- * Uzyskanie wartości określonej komórki:

```
tablica[rzqd][kolumna];
```

Umownie - można by zapisać
raczej: wymiar1, wymiar2...

- * Np:

```
System.out.println(tablica[6][4]);
```

	0	1	2	3	4	5	6	7	8	9
0	0	1	2	3	4	5	6	7	8	9
1	10	11	12	13	14	15	16	17	18	19
2	20	21	22	23	24	25	26	27	28	29
3	30	31	32	33	34	35	36	37	38	39
4	40	41	42	43	44	45	46	47	48	49
5	50	51	52	53	54	55	56	57	58	59
6	60	61	62	63	64	65	66	67	68	69
7	70	71	72	73	74	75	76	77	78	79
8	80	81	82	83	84	85	86	87	88	89
9	90	91	92	93	94	95	96	97	98	99

Tablice wielowymiarowe

- * Wypisanie zawartości tabeli w 10 rzędach po 10 liczb:


```
for (int i=0; i<10; i++) {  
    for (int j=0; j<10; j++ ) {  
        System.out.print(tablica[i][j]+" ");  
    }  
    System.out.println();  
}
```

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29
30	31	32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47	48	49
50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69
70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89
90	91	92	93	94	95	96	97	98	99

Wypełnianie tablic wielowymiarowych gotowymi wartościami


- * Podobnie jak tablicę jednowymiarową, tablice wielowymiarowe można wypełnić wartościami od razu po deklaracji:

Deklaracja, utworzenie i wypełnienie wartościami



```
int[][] tab = {{1,2,3,4,5}, {6,7,8,9,10}, {11,12,13,14,15}};
```

```
for (int[] l : tab) {  
    for (int m : l) System.out.print(m+" ");  
    System.out.println();  
}
```



Sprawdzamy (drukujemy zawartość tablicy)

Wykorzystanie **foreach** w tablicy wielowymiarowej

W pierwszej pętli
pobierane są tablice 1
wymiarowe

```
for (int[] t : tablica) {  
    for (int l : t) {  
        System.out.print(l+" ");  
    }  
}
```

W drugiej, zagnieżdżonej,
pętli pobierane są kolejne
wartości z tablicy **t**

Tablice wielowymiarowe

- * Tablice mogą mieć 3, 4 i więcej wymiarów. Deklarujemy, tworzymy i pracujemy na nich analogicznie do tablic 2-wymiarowych. Np:

```
int[][][] tablica3wymiarowa = new int[5][10][10];
```

```
for (int i=0; i<5; i++) {  
    for (int j=0; j<10; j++ ) {  
        for (int k=0; k<10; k++)  
            tablica3wymiarowa[i][j][k] = i+j+k;  
    }  
}
```

```
for (int[][] t : tablica3wymiarowa) {  
    for (int[] u : t) {  
        for (int w : u) System.out.print(w+" ");  
        System.out.println();  
    }  
    System.out.println();  
    System.out.println();  
}
```

Zadanie

- * Napisz program który znajdzie największą liczbę w tablicy 2 wymiarowej zawierającej liczby nieujemne.

Zadanie

- * Napisz program który znajdzie największą liczbę w tablicy 2 wymiarowej zawierającej liczby nieujemne.

```
int[][] tab = {{1,2,3,4,5},
               {6,7,8,9,10},
               {11,22,13,14,15}};
int najwiekszaLiczba = 0;
for (int[] l : tab) {
    for (int m : l) {
        if (najwiekszaLiczba < m)
            najwiekszaLiczba = m;
    }
}
```

```
System.out.println("Największa liczba to "+najwiekszaLiczba);
```