

Pętle FOR i macierze

Wpisany przez Administrator

Z tej lekcji dowiesz się:

- co to są macierze
- kiedy stosujemy macierze
- co to są pętle (iteracje)
- kiedy stosujemy pętle
- jak stosować pętlę "for"

Nauczysz się:

- stosować macierze w języku Java
- stosować pętlę "for" w języku Java

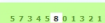
Co to są macierze?

Zobacz poniższy przykład:

5 7 3 4 5 8 0 1 3 2 1

To, co wypisane jest powyżej to zbiór cyfr. Co więcej, zbiór ten jest uporządkowany, czyli każdy element tego zbioru ma swoją wartość ale także ma swoje miejsce w kolejności.

Np. element o indeksie 6 (indeks to numer obiektu) ma wartość 8.



5 7 3 4 5 8 0 1 3 2 1

Rozmiar tej macierzy wynosi 11 (tyle elementów ona zmieści)

Jest to przykład **macierzy 1-wymiarowej**.

Skoro jest macierz 1-wymiarowa to pewnie jest też 2-wymiarowa. Poniżej przedstawiam przykład:

Pętle FOR i macierze

Wpisany przez Administrator

6 3 6 3 7 1 3 9 6

3 4 8 3 4 6 1 8 9

6 8 0 3 5 6 7 8 9

1 7 9 3 6 8 9 1 0

Poniższa macierz ma wymiary 9 na 4 (9 elementów w rzędzie razy cztery rzędy).

Każdy z elementów można wskazać podając jego współrzędne, np element o współrzędnych [5,3] ma wartość 5.

6	3	6	3	7	1	3	9	6
3	4	8	3	4	6	1	8	9
6	8	0	3	5	6	7	8	9
1	7	9	3	6	8	9	1	0

Zauważ, że w macierzy 2-wymiarowej potrzebne są aż 2 indeksy. (indeks kolumny i indeks wiersza lub odwrotnie, zależnie od przyjętego kanonu)

Zauważ też, że macierz 2 wymiarowa musi być regularna tzn., że wszystkie wiersze muszą być

tej samej długości.

Macierze wykorzystywane są do przechowywania zbioru danych tego samego typu (u nas są to liczby całkowite) w sposób uporządkowany (czyli taki, gdzie każdy element ma swoje miejsce, swój adres czy swoje współrzędne).

W programowaniu macierze wykorzystywane są bardzo często, jednym z najbardziej rozpoznawalnych przykładów może być arkusz kalkulacyjny, którego działanie opiera się na macierzy.

Obsługa macierzy (szczególnie tych dużych, które nieraz liczą tysiące elementów mogłaby być bardzo żmudna i podatna na błędy.

Wyobraź sobie sytuację, w której macierz 10-elementową musisz wyzerować (wypełnić elementami o wartości 0). Pisząc kod "na piechotę" musielibyśmy napisać 10 linijek kodu, w każdej linijce kolejny element byłby zerowany. Zobaczmy to na przykładzie:

Uruchom program **NetBeans**

Klikamy kolejno **File > New Project**

W oknie wybieramy Odpowiednio **Java** oraz **Java Application**

Klikamy na **Next**

Podajemy nazwę programu: **matrix** i klikamy na **Finish**

W głównej klasie programu wpisujemy następujący kod:

```
1  /*
2   * To change this template, choose Tools | Templates
3   * and open the template in the editor.
4   */
5  package matrix;
6
7  /**
8   *
9   * @author ticzex
10  */
11 public class Matrix {
12
13     /**
14      * @param args the command line arguments
15      */
16     public static void main(String[] args) {
17         // TODO code application logic here
18         int[] matrix1 = new int[10];
19         matrix1[0]=0;
20         matrix1[1]=0;
21         matrix1[2]=0;
22         matrix1[3]=0;
23         matrix1[4]=0;
24         matrix1[5]=0;
25         matrix1[6]=0;
26         matrix1[7]=0;
27         matrix1[8]=0;
28         matrix1[9]=0;
29     }
30 }
31
```

Właśnie nauczyliśmy się deklarować zmienną tablicową (linijka 18) oraz odwoływać do poszczególnych elementów tablicy (linijki 19-28).

Oczywiście zamiast liczb całkowitych (int) tablica może zawierać elementy innego typu (np.: liczby zmiennoprzecinkowe (float) lub ciągi znaków (napisy: String), lub choćby nawet inne tablice, ale o tym kiedy indziej.

Teraz sprawdzimy, czy program działa; każmy mu wyświetlić wartość dowolnego elementu,, np.

Pętle FOR i macierze

Wpisany przez Administrator

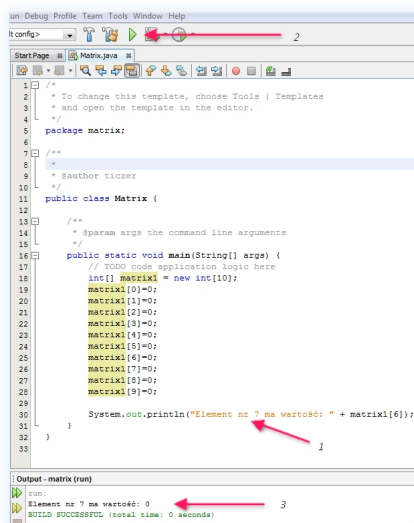
element 7 w kolejności:

W tym celu:

1) W linii nr 30 wpisujemy komendę wyświetlającą informację o elemencie o indeksie 6 (licząc z zerem jest to siódmy element)

2) Klikamy na zieloną strzałkę, która uruchamia program

3) Zaprogramowany komunikat pojawia się w dolnej części okna



Aby kod programu zajmował mniej miejsca usuwamy komentarze ("wyszarzone" napisy znajdujące się między znakami `/*` a `*/`)

Pętle FOR i macierze

Wpisany przez Administrator

```
1 package matrix;
2 public class Matrix {
3
4     public static void main(String[] args) {
5         // TODO code application logic here
6         int[] matrix1 = new int[10];
7         matrix1[0]=0;
8         matrix1[1]=0;
9         matrix1[2]=0;
10        matrix1[3]=0;
11        matrix1[4]=0;
12        matrix1[5]=0;
13        matrix1[6]=0;
14        matrix1[7]=0;
15        matrix1[8]=0;
16        matrix1[9]=0;
17
18        System.out.println("Element nr 7 ma wartość: " + matrix1[6]);
19    }
20 }
21
```

Kod źródłowy programu, który nie wykorzystuje pętli. Wykonując go, otrzymamy jedno zdanie: "Element nr 7 ma wartość: 0".

```
1 package matrix;
2 public class Matrix {
3
4     public static void main(String[] args) {
5         // TODO code application logic here
6         int[] matrix1 = new int[10];
7         for (int i = 0; i < 10; i++) {
8             matrix1[i] = 0;
9         }
10        System.out.println("Element nr 7 ma wartość: " + matrix1[6]);
11    }
12 }
13
```

Kod źródłowy programu, który wykorzystuje pętlę for do przypisania wartości 0 do wszystkich elementów tablicy. Wykonując go, otrzymamy to samo zdanie: "Element nr 7 ma wartość: 0".

```
1 package matrix;
2 public class Matrix {
3
4     public static void main(String[] args) {
5         // TODO code application logic here
6         int[] matrix1 = new int[10];
7         for (int i = 0; i < 10; i++) {
8             matrix1[i] = 0;
9         }
10        for (int k = 0; k < 10; k++) {
11            System.out.println("Element nr " + k + " ma wartość: " + matrix1[k]);
12        }
13    }
14 }

```

Output - matrix (run)

```
run:
Element nr 0 ma wartość: 0
Element nr 1 ma wartość: 0
Element nr 2 ma wartość: 0
Element nr 3 ma wartość: 0
Element nr 4 ma wartość: 0
Element nr 5 ma wartość: 0
Element nr 6 ma wartość: 0
Element nr 7 ma wartość: 0
Element nr 8 ma wartość: 0
Element nr 9 ma wartość: 0
BUILD SUCCESSFUL (total time: 1 second)
```

W tym celu możemy skorzystać z metody Arrays.fill(), która umożliwia wypełnienie tablicy wartością 0. Napisz program, który wypisze wszystkie elementy tablicy, które zostały wypełnione wartością 0.