

Wstęp

Tablice są jedną z najczęściej wykorzystywanych struktur danych w programowaniu. Są statycznymi strukturami danych - oznacza to, że musimy w trakcie deklaracji podać ich wymiary. Pojemność tablicy będzie jednoznacznie określona w trakcie całego programu (tj. nie możemy np. usuwać z nich elementy na zasadzie zwalniania pamięci ze sterty). Tablice są referencyjnym typem danych – oznacza to, że przekazywane są np. do innych metod poprzez referencję – czyli wskaźnik na nie. Tablice jako typ referencyjny są, tak samo wszystkie inne struktury danych w JAVIE, przechowywane na stercie.

Np.

Poniższą deklarację Tablicy jednowymiarowej, należy czytać (rozumieć):

```
int [] Tab = new int[3];
```

Na Stosie (zakładając deklarację tablicy w jakiejś metodzie) zostanie odłożony wskaźnik Tab, który będzie wskazywać na adres ze Sterty, pod którym kolejno będą zapisane 3 wartości całkowite.

Ponieważ tablica jest typem referencyjnym, jeśli ją prześlemy do innej metody, to nie będziemy działać na jej 'kopii', tylko na niej samej, ponieważ w istocie przekazujemy wartość jednego wskaźnika (adres tablicy), który jest przechwytywany przez inny wskaźnik.

Tablica dwuwymiarowa jest interpretowana jako tablica tablic – tj. jednowymiarowa tablica wskaźników na kolejne tablice jednowymiarowe. Dzięki temu możemy bez problemu np. definiować tablice o zmiennej liczbie kolumn. Tablicę dwuwymiarową interpretujemy jako macierz. Poniżej przykład klasycznego przejścia przez elementy tablicy dwuwymiarowej za pomocą dwóch zagnieżdżonych pętli:

„ZPR PWR – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

```
int [][] Tab = new int[2][3];

Tab[0][0] = 10; Tab[0][1] = 4; Tab[0][2] = 10;
Tab[1][0] = 4; Tab[1][1] = 5; Tab[1][2] = 60;

for(int i=0; i<Tab.length; i++)
{
    for(int j=0; j<Tab[0].length; j++)
        System.out.print("Element Tab[" + i + "]" + "[" + j + "] = " + Tab[i][j] + "\t");

    System.out.println("\n\n");
}
```

Wynik działania takiego kodu (**Console**):

Element Tab[0][0] = 10 Element Tab[0][1] = 4 Element Tab[0][2] = 10

Element Tab[1][0] = 4 Element Tab[1][1] = 5 Element Tab[1][2] = 60

Dodatkowo jako załącznik na ostatniej stronie zaprezentowano rozwiązanie zadania 4.

„ZPR PWr – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

Lista 4

1. (kod Java)

Zaprezentuj funkcje, za pomocą których można:

- wygenerować losowe wartości (*int*) dla 100-elementowej tablicy jednowymiarowej (rozpoznaj i wykorzystaj klasę *Random*),
- wyszukać wartość maksymalną w tablicy,
- wyszukać wartość minimalną w tablicy,
- określić wartość średnią w tablicy (średnia arytmetyczna),
- wyszukać tylko wartości parzyste oraz wpisać je do osobnej tablicy.

2. (schemat Blokowy)

Przedstaw schemat blokowy dla algorytmu mnożenia dwóch macierzy.

3. (kod Java)

Zaprezentuj funkcję, która wykonuje mnożenie dwóch macierzy.

4. (kod Java)

Zdefiniuj tablicę dwuwymiarową o liczbie wierszy 5 i zmiennej liczbie kolumn:

- wiersz 1: 4 kolumny,
- wiersz 2: 2 kolumny,
- wiersz 3: 3 kolumny,
- wiersz 4: 2 kolumny,
- wiersz 5: 5 kolumny.

Zaprezentuj funkcję, za pomocą której wyszukasz wartość maksymalną w tak określonej tablicy.

Uwaga: zastanów się jak przekazać tablicę do funkcji (metody). *Nazwa tablicy to wskaźnik, który ...*

```
package Tablica2D_przykład;

import java.util.Random;

public class Table2D_example {

    final static int w = 5;

    public static void main(String[] args)
    {
        //Generujemy tablice o różnej liczbie kolumn
        double [][] Tab = new double[w][];

        Tab[0] = new double[4];
        Tab[1] = new double[2];
        Tab[2] = new double[3];
        Tab[3] = new double[2];
        Tab[4] = new double[5];

        Table2D_example obT = new Table2D_example();
        obT.wypelnijMacierz(Tab);
        obT.drukujMacierz(Tab);

        System.out.println("\n\nMax Tablicy = " + obT.szukajMax(Tab));
    }

    //Wypełniamy losowo tablicę liczbami typu double z zakresu 0 - 10
    public void wypelnijMacierz(double [][] T)
    {
```

„ZPR PWr – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

```
Random generator = new Random();

for(int i=0; i<T.length; i++)
    for(int j=0; j<T[i].length; j++)
        T[i][j] = generator.nextDouble()*10;

}

//Wyświetlamy wartości tablicy
public void drukujMacierz(double [][] T)
{
    for(int i=0; i<T.length; i++)
    {
        for(int j=0; j<T[i].length; j++)
            System.out.print("Tab[" + i + "]" + "[" + j + "]" + "=" + T[i][j] + "\t" );

        System.out.println();
    }
}

//Znajdujemy max
public double szukajMax(double [][] T)
{
    double Max = T[0][0];

    for(int i=0; i<T.length; i++)
        for(int j=0; j<T[i].length; j++)
            if (T[i][j] > Max) Max = T[i][j];

    return Max;
}
```

„ZPR PWr – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

}

Console:

```
Tab[0][0]=6.044309682151625  Tab[0][1]=0.6116608046090977  Tab[0][2]=5.5378806833526575  Tab[0][3]=9.565020958279103
Tab[1][0]=9.484942038499993  Tab[1][1]=5.137896603922634
Tab[2][0]=5.619902650861813  Tab[2][1]=3.2530344926035184  Tab[2][2]=9.225801479823387
Tab[3][0]=4.093873787462499  Tab[3][1]=6.703191683860022
Tab[4][0]=4.333331936847239  Tab[4][1]=6.278305916070151  Tab[4][2]=8.984443401321109  Tab[4][3]=1.8895716275910734
    Tab[4][4]=1.9909351503352624
```

Max Tablicy = 9.565020958279103