

Національний технічний університет

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки,

Лабораторна робота № 4

|  |
| --- |
| Виконав: |
| студент 3 курсу ФІОТ |
| Групи ІО-91  Матвійчук Б.В. |
|  |

Київ-2011

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

namespace pro\_lab4

{

/\*\* головний класc

\* @author Матвійчук Богдан группа ІО-91

\* 1.7 b = (A\*SORT(C))

\* 2.20 MD = MA + MB – MC

\* 3.18 p = MAX(SORT(MS) + MA\*MB)

\*/

class MAIN

{

static void Main(){

Thread.CurrentThread.Name = "MAIN";

Console.WriteLine(Thread.CurrentThread.Name + "start");

f1 fun1 = new f1();

Thread f1t = new Thread(fun1.Go);

f1t.Name = "Fun1";

f1t.Start();

f2 fun2 = new f2();

Thread f2t = new Thread(fun2.Go);

f2t.Name = "Fun2";

f2t.Start();

f3 fun3 = new f3();

Thread f3t = new Thread(fun3.Go);

f3t.Name = "Fun3";

f3t.Start();

f1t.Join();

f3t.Join();

f2t.Join();

Console.ReadKey();

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

namespace pro\_lab4

{

/\*\*

\* @author Матвійчук Богдан группа ІО-91

\* 1.7 b = (A\*SORT(C))

\*/

public class f1{

public void Go (){

Console.WriteLine(Thread.CurrentThread.Name+ "Start");

int [] A = new int[Package.size];

Package.set1Vector(A);

Console.WriteLine("imput A");

int [] C = new int[Package.size];

Package.set1Vector(C);

Console.WriteLine("imput C");

int pak = Package.mult(A, Package.sortVector (C));

Console.WriteLine ("b= " + pak);

Console.WriteLine(Thread.CurrentThread.Name+ "end");

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

namespace pro\_lab4

{

class f2

{

public void Go()

{

Console.WriteLine(Thread.CurrentThread.Name + "Start");

int[,] MA = new int[Package.size, Package.size];

Package.set1Matrix(MA);

Console.WriteLine("imput MA");

int[,] MB = new int[Package.size, Package.size];

Package.set1Matrix(MB);

Console.WriteLine("imput MB");

int[,] MC = new int[Package.size, Package.size];

Package.set1Matrix(MC);

Console.WriteLine("imput MC");

int[,] MD = new int[Package.size, Package.size];

MD = Package.minus(Package.Plus(MA, MB), MC);

if (Package.size < 10)

{

Console.WriteLine("MD= ");

Console.WriteLine(Package.toStringMatrix(MD));

}

Console.WriteLine(Thread.CurrentThread.Name + "end");

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

namespace pro\_lab4

{

class f3

{

public void Go()

{

Console.WriteLine(Thread.CurrentThread.Name + "Start");

int[,] MA = new int[Package.size, Package.size];

Package.set1Matrix(MA);

Console.WriteLine("imput MA");

int[,] MB = new int[Package.size, Package.size];

Package.set1Matrix(MB);

Console.WriteLine("imput MB");

int[,] MS = new int[Package.size, Package.size];

Package.set1Matrix(MS);

Console.WriteLine("imput MS");

int p = Package.Max(Package.Plus(MS, Package.mult(MA, MB)));

Console.WriteLine("p= " + p);

Console.WriteLine(Thread.CurrentThread.Name + "end");

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace pro\_lab4

{

/\*\* класc з функціями

\* @author Матвійчук Богдан группа ІО-91

\* 1.7 b = (A\*SORT(C))

\* 2.20 MD = MA + MB – MC

\* 3.18 p = MAX(SORT(MS) + MA\*MB)

\*/

public class Package

{

/\*\*

розмір (вектора/масива)

\*/

public static int size = 1000;

/\*\* задає значення вказаной перемінной

\* @param n номер позиції mas[n][m]

\* @param m номер позиції mas[n][m]

\* @param val перемінна int

\*/

public static void setMatrix(int[,] arr, int n, int m, int val)

{

arr[n, m] = val;

}

/\*\*

\* заповнює матрицю 1

\*/

public static void set1Matrix(int[,] mas)

{

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

mas[i, j] = 1;

}

}

}

public static void set1Vector(int[] arr)

{

for (int i = 0; i < arr.Length; i++)

{

arr[i] = 1;

}

}

/\*\* розмір матриці

\* @return int

\*/

public static int getSize(int[,] arr)

{

int a = arr.Length;

return a;

}

/\*\* формування строкі

\* @return String

\*/

public static String toStringMatrix(int[,] arr)

{

String res = "";

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

res += arr[i, j];

}

res += '\n';

}

return res;

}

/\*\* вертає перемінну

\* @param index номер позиції перемінной int

\* @return int

\*/

public static int getVector(int[] arr, int index)

{

return arr[index];

}

/\*\* розмір вектора

\* @return int

\*/

public static int getSizeVector(int[] arr)

{

return arr.Length;

}

/\*\* формування строкі

\* @return String

\*/

public static String toStringVector(int[] arr)

{

String res = "";

for (int i = 0; i < arr.Length; i++)

{

res += " " + arr[i];

}

return res;

}

/\*\* A\*B

\* @param A Vector

\* @param B Vector

\* @return int

\*/

public static int mult(int[] A, int[] B)

{

int R = 0;

for (int i = 0; i < A.Length; i++)

{

R = R + A[i] \* B[i];

}

return R;

}

/\*\* Max Vector

\* @param vect Vector

\* @return Vector

\*/

public static int Max(int[] vect)

{

int Max = vect[0];

for (int i = 0; i < vect.Length; i++)

{

if (Max < vect[i])

{

Max = vect[i];

}

}

return Max;

}

/\*\* MA+MB

\* @param MA matrix

\* @param MB matrix

\* @return matrix

\*/

public static int[,] Plus(int[,] MA, int[,] MB)

{

int[,] MC = new int[size, size];

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

MC[i, j] = MA[i, j] + MB[i, j];

}

}

return MC;

}

/\*\* MA-MB

\* @param MA matrix

\* @param MB matrix

\* @return matrix

\*/

public static int[,] minus(int[,] MA, int[,] MB)

{

int[,] MC = new int[size, size];

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

MC[i, j] = MA[i, j] - MB[i, j];

}

}

return MC;

}

/\*\* Max matrix

\* @param matrix matrix

\* @return matrix

\*/

public static int Max(int[,] matrix)

{

int Max = matrix[0, 0];

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (Max < matrix[i, j])

{

Max = matrix[i, j];

}

}

}

return Max;

}

/\*\* MA\*MB

\* @param MA matrix

\* @param MB matrix

\* @return matrix

\*/

public static int[,] mult(int[,] MA, int[,] MB)

{

int[,] Result = new int[size, size];

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

Result[i, j] = 0;

for (int y = 0; y < size; y++)

{

Result[i, j] = Result[i, j] + MA[i, y] \* MB[y, j];

}

}

}

return Result;

}

/\*\* Sort Vector

\* @param V Vector

\* @return Vector

\*/

public static int[] sortVector(int[] V)

{

int z;

for (int i = 0; i < V.Length; i++)

{

for (int j = 0; j < V.Length - 1; j++)

{

if (V[j] > V[j + 1])

{

z = V[j];

V[j] = V[j + 1];

V[j + 1] = z;

}

}

}

return V;

}

/\*\* Sort Matrix

\* @param MA matrix

\* @return matrix

\*/

public static int[,] sortMatrix(int[,] MA)

{

int z;

for (int i = 0; i < MA.Length; i++)

{

for (int j = 0; j < MA.Length - 1; j++)

{

for (int k = 0; k < MA.Length - 2; j++)

{

if (MA[j, k] > MA[j, k + 1])

{

z = MA[j, k];

MA[j, k] = MA[j, k + 1];

MA[j, k + 1] = z;

}

}

}

}

return MA;

}

/\*\* 2.20 MD = MA + MB – MC

\* @param MA matrix

\* @param MB matrix

\* @param MC matrix

\* @return matrix

\*/

public int[,] fun2(int[,] MA, int[,] MB, int[,] MC)

{

return minus(Plus(MA, MB), MC);

}

/\*\* 3.18 p = MAX(SORT(MS) + MA\*MB)

\* @param MS matrix

\* @param MA matrix

\* @param MB matrix

\* @return int

\*/

public int fun3(int[,] MS, int[,] MA, int[,] MB)

{

return Max(Plus(MS, mult(MA, MB)));

}

}

}