

Національний технічний університет

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки,

Лабораторна робота № 3

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

Київ-2011

**package** lab3;

/\*\* головний клас

\* **@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** main {

**public** **static** **void** main (String[] args) {

Thread1 t1 = **new** Thread1(7);

Thread2 t2 = **new** Thread2(9);

Thread3 t3 = **new** Thread3(6);

t1.start();

**try** {

t1.join();

} **catch** (InterruptedException e) {

e.printStackTrace();

}

t2.start();

**try** {

t2.*sleep*(100);

} **catch** (InterruptedException e) {

e.printStackTrace();

}

t3.start();

}

}

**package** lab3;

**import** java.util.Scanner;

/\*\* 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*=3000;

/\*\* imput key

\* **@return** j

\*/

**public** **static** **int** inp() {

Scanner input = **new** Scanner(System.*in*);

**int** i = input.nextInt();

// int i = 1;

**return** i;

}

/\*\* A\*B

\* **@param** A Vector

\* **@param** B Vector

\* **@return** int

\*/

**private** **static** **int** mult(Vector A, Vector B) {

**int** R = 0;

**for** (**int** i = 0; i < A.getSize(); i++) {

R = R + A.get(i) \* B.get(i);

}

**return** R;

}

/\*\* Max Vector

\* **@param** vect Vector

\* **@return** Vector

\*/

**private** **static** **int** Max(Vector vect) {

**int** Max = vect.get(0);

**for** (**int** i = 0; i < vect.getSize(); i++) {

**if** (Max < vect.get(i)) {

Max = vect.get(i);

}

}

**return** Max;

}

/\*\* MA+MB

\* **@param** MA matrix

\* **@param** MB matrix

\* **@return** matrix

\*/

**private** **static** matrix plus(matrix MA, matrix MB) {

matrix MC = **new** matrix(MA.getSize(), MA.getSize());

**for** (**int** i = 0; i < MA.getSize(); i++) {

**for** (**int** j = i; j < MA.getSize(); j++) {

MC.set(i, j, MA.get(i, j) + MB.get(i, j));

}

}

**return** MC;

}

/\*\* MA-MB

\* **@param** MA matrix

\* **@param** MB matrix

\* **@return** matrix

\*/

**private** **static** matrix minus(matrix MA, matrix MB) {

matrix MC = **new** matrix(MA.getSize(), MA.getSize());

**for** (**int** i = 0; i < MA.getSize(); i++) {

**for** (**int** j = i; j < MA.getSize(); j++) {

MC.set(i, j, MA.get(i, j) - MB.get(i, j));

}

}

**return** MC;

}

/\*\* Max matrix

\* **@param** matrix matrix

\* **@return** matrix

\*/

**private** **static** **int** Max(matrix matrix) {

**int** Max = matrix.get(0,0);

**for** (**int** i = 0; i < matrix.getSize(); i++) {

**for** (**int** j = 0; j < matrix.getSize(); j++) {

**if** (Max < matrix.get(i,j)) {

Max = matrix.get(i,j);

}

}

}

**return** Max;

}

/\*\* MA\*MB

\* **@param** MA matrix

\* **@param** MB matrix

\* **@return** matrix

\*/

**private** **static** matrix mult(matrix MA, matrix MB) {

matrix Result = **new** matrix(MA.getSize(), MA.getSize());

**for** (**int** i = 0; i < MA.getSize(); i++) {

**for** (**int** j = 0; j < MA.getSize(); j++) {

Result.set(i, j, 0);

**for** (**int** y = 0; y < MA.getSize(); y++) {

Result.set(i, j, Result.get(i, j) + MA.get(i, y)

\* MB.get(y, j));

}

}

}

**return** Result;

}

/\*\* Sort Vector

\* **@param** V Vector

\* **@return** Vector

\*/

**private** Vector sortVector(Vector V) {

**int** z;

**for** (**int** i = 0; i < V.getSize(); i++) {

**for** (**int** j = 0; j < V.getSize() - 1; j++) {

**if** (V.get(j) > V.get(j + 1)) {

z = V.get(j);

V.set(j, V.get(j + 1));

V.set((j + 1), z);

}

}

}

**return** V;

}

/\*\* Sort Matrix

\* **@param** MA matrix

\* **@return** matrix

\*/

**private** matrix sortMatrix(matrix MA) {

**int** z;

**for** (**int** i = 0; i < MA.getSize(); i++) {

**for** (**int** j = 0; j < MA.getSize() - 1; j++) {

**for** (**int** k = 0; k < MA.getSize() - 2; j++) {

**if** (MA.get(j, k) > MA.get(j, k + 1)) {

z = MA.get(j, k);

MA.set(j, k, MA.get(j, k + 1));

MA.set(j, k + 1, z);

}

}

}

}

**return** MA;

}

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

\* **@param** A Vector

\* **@param** C Vector

\* **@return** int

\*/

**public** **int** fun1 (Vector A,Vector C){

**return** *mult*(A, sortVector(C));

}

/\*\* 2.20 MD = MA + MB – MC

\* **@param** MA matrix

\* **@param** MB matrix

\* **@param** MC matrix

\* **@return** matrix

\*/

**public** matrix fun2 (matrix MA, matrix MB, matrix 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 (matrix MS, matrix MA, matrix MB){

**return** *Max*(*plus*(MS, *mult*(MA, MB)));

}

}

**package** lab3;

**public** **class** matrix {

/\*\*

\* матриця int

\*/

**private** **int**[][] arr;

/\*\*

\* розмір матриці mas[n][m]

\*/

**private** **int** n;

/\*\*

\* розмір матриці mas[n][m]

\*/

**private** **int** m;

/\*\*

\* **@param** n розмір матриці mas[n][m]

\* **@param** m розмір матриці mas[n][m]

\*/

**public** matrix(**int** n, **int** m) {

arr = **new** **int** [n][m];

**this**.n=n;

**this**.m=m;

}

/\*\* задає значення вказаной перемінной

\* **@param** n номер позиції mas[n][m]

\* **@param** m номер позиції mas[n][m]

\* **@param** val перемінна int

\*/

**public** **void** set(**int** n, **int** m, **int** val) {

arr[n][m]=val;

}

/\*\*

\* заповнює матрицю 1

\*/

**public** **void** set1() {

**for** (**int** i=0;i<n;i++){

**for** (**int** j=0;j<m-1;j++){

arr[i][j]=1;

}

}

}

/\*\* вертає перемінну

\* **@param** index n номер позиції перемінной mas[n][m]

\* **@param** index m номер позиції перемінной mas[n][m]

\* **@return** int

\*/

**public** **int** get(**int** a, **int** b) {

**return** arr[a][b];

}

/\*\* розмір матриці

\* **@return** int

\*/

**public** **int** getSize() {

**return** arr.length;

}

/\*\* формування строкі

\* **@return** String

\*/

**public** String toString() {

String res = "";

**for** (**int** i=0;i<n;i++){

**for** (**int** j=0;j<m;j++){

res+=Integer.*toString*(arr[n][m]);

}

res+='\n';

}

**return** res;

}

}

**package** lab3;

/\*\* реалізує функції для вектора

\* **@author** Матвійчук Богдан группа ІО-91

\*

\*/

**public** **class** Vector {

/\*\*

\* вектор int

\*/

**private** **int**[] arr;

/\*\*

\* **@param** n розмір вектора

\*/

**public** Vector(**int** n) {

arr = **new** **int**[n];

}

/\*\*

\* заповнює вектор 1

\*/

**public** **void** set1() {

**for** (**int** i=0;i<arr.length;i++){

arr[i]=1;

}

}

/\*\* задає значення вказаной перемінной

\* **@param** index номер позиції int

\* **@param** val перемінна int

\*/

**public** **void** set(**int** index, **int** val) {

arr[index] = val;

}

/\*\* вертає перемінну

\* **@param** index номер позиції перемінной int

\* **@return** int

\*/

**public** **int** get(**int** index) {

**return** arr[index];

}

/\*\* розмір вектора

\* **@return** int

\*/

**public** **int** getSize() {

**return** arr.length;

}

/\*\* формування строкі

\* **@return** String

\*/

**public** String toString() {

String res = "";

**for** (**int** i = 0; i < arr.length; i++){

res += " " + arr[i];}

**return** res;

}

}

**package** lab3;

/\*\*

\* **@author** Матвійчук Богдан группа ІО-91

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

\*/

**public** **class** Thread1 **extends** Thread {

**public** Thread1(**int** prior) {

setPriority(prior);

}

**public** **void** run (){

Package pak = **new** Package ();

System.*out*.println("T1 started");

Vector A = **new** Vector(Package.*size*);

Vector C = **new** Vector(Package.*size*);

System.*out*.println(" Input A");A.set1();

System.*out*.println(" Input C");C.set1();

**int** b = pak.fun1(A, C);

System.*out*.println(b);

System.*out*.println("T1 finished");

}

}

**package** lab3;

/\*\*

\* **@author** Матвійчук Богдан группа ІО-91

\* 2.20 MD = MA + MB – MC

\*/

**public** **class** Thread2 **extends** Thread {

**public** Thread2(**int** prior) {

setPriority(prior);

}

**public** **void** run (){

Package pak = **new** Package ();

System.*out*.println("T2 started");

matrix MA = **new** matrix (Package.*size*,Package.*size*);

matrix MB = **new** matrix (Package.*size*,Package.*size*);

matrix MC = **new** matrix (Package.*size*,Package.*size*);

matrix MD = **new** matrix (Package.*size*,Package.*size*);

System.*out*.println(" Input MA");MA.set1();

System.*out*.println(" Input MB");MB.set1();

System.*out*.println(" Input MC");MC.set1();

MD.set1();

MD = pak.fun2(MA, MB, MC);

if (Package.size<10){

System.out.println(MD.toString());

}

System.*out*.println("T2 finished");

}

}

**package** lab3;

/\*\*

\* \* **@author** Матвійчук Богдан группа ІО-91

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

\*

\*/

**public** **class** Thread3 **extends** Thread {

**public** Thread3(**int** prior) {

setPriority(prior);

}

**public** **void** run (){

Package pak = **new** Package ();

System.*out*.println("T3 started");

matrix MA = **new** matrix (Package.*size*,Package.*size*);

matrix MB = **new** matrix (Package.*size*,Package.*size*);

matrix MS = **new** matrix (Package.*size*,Package.*size*);

System.*out*.println(" Input MA");MA.set1();

System.*out*.println(" Input MB");MB.set1();

System.*out*.println(" Input MC");MS.set1();

**int** p = pak.fun3(MS, MA, MB);

System.*out*.println(p);

System.*out*.println("T3 finished");

}

}