Национальный технический университет

«Киевский политехнический институт»

Факультет информатики и вычислительной техники

## Кафедра Вычислительной Техники

#### Лабораторная работа №4

По курсу: Параллельное программирование

студента III-го курса,

группы ИВ-92

Петрука В.О.

## Киев 2011 г.

/\*Laboratory work 4

Petruk Vadim, IO-92

Variant: 1.13 C = A - B + D

2.28 MD = MIN(MA)\*MB\*MC

3.14 D = (A + B)\*(MA - MB)

\*/

using System;

using System.Collections.Generic;

using System.Text;

using System.Threading;

namespace Lab4\_vad

{

class Program

{

private static int N = 3000;

static void Main(string[] args)

{

Thread tread1 = new Thread(Function1);

Thread tread2 = new Thread(Function2);

Thread tread3 = new Thread(Function3);

tread1.Priority = ThreadPriority.Highest;

tread2.Priority = ThreadPriority.Normal;

tread3.Priority = ThreadPriority.Lowest;

tread1.Start();

tread2.Start();

tread3.Start();

Console.ReadLine();

}

static void Function1()

{

Console.WriteLine("Function1 is started");

int[] A = VectorIn(N);

int[] B = VectorIn(N);

int[] C = new int[N];

int[] D = VectorIn(N);

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

{

C[i] = A[i] - B[i] + D[i];

}

if (N<=8)

{

VectorOut(C);

}

Thread.Sleep(1000);

Console.WriteLine("Function1 is finished");

}

static void Function2()

{

Console.WriteLine("Function2 is started");

int[,] MA = MatrixIn(N);

int[,] MB = MatrixIn(N);

int[,] MC = MatrixIn(N);

int[,] MD = new int[N,N];

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

{

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

{

MD[i,j]=0;

}

}

//MD := MB\*MC

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

{

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

{

for (int k=0; k<N; k++)

{

MD[i,j] = MD[i,j]+MB[i,k]\*MC[k,j];

}

}

}

//search MIN(MA)

int min\_MA = MA[0,0];

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

{

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

{

if (MA[i,j] < min\_MA)

{

min\_MA = MA[i,j];

}

}

}

//MD := MIN(MA)\*MB\*MC

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

{

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

{

MD[i,j]=MD[i,j]\*min\_MA;

}

}

if (N<=8)

{

MatrixOut(MD);

}

Console.WriteLine("Function2 is finished");

}

//Func3: D := (A + B)\*(MA - MB)

static void Function3()

{

Console.WriteLine("Function3 is started");

int[] A = VectorIn(N);

int[] B = VectorIn(N);

int[,] MA = MatrixIn(N);

int[,] MB = MatrixIn(N);

int[] D = new int[N];

int[] C = new int[N];

//C:=A + B

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

{

C[i] = A[i]+B[i];

}

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

//MC:=MA - MB

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

{

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

{

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

}

}

//MD = 0

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

{

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

{

D[i]=0;

}

}

//D := C \* MC

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

{

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

{

D[j] = D[j] + C[j]\*MC[j,i];

}

}

if (N<=8)

{

VectorOut(D);

}

Console.WriteLine("Function3 is finished");

}

static int[] VectorIn(int N)

{

int[] X = new int[N];

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

{

X[i] = 1;

}

return X;

}

static void VectorOut(int[] X)

{

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

{

Console.Write(X[i]+" ");

}

Console.WriteLine();

}

static int[,] MatrixIn(int N)

{

int[,] MX = new int[N,N];

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

{

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

{

MX[i,j] = 1;

}

}

return MX;

}

static void MatrixOut(int[,] MX)

{

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

{

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

{

Console.Write(MX[i,j]+" ");

if (j==N-1)

Console.WriteLine();

}

}

}

}

}