Лабораторная работа №5

по курсу:

«Паралельные и распределённые вычисления»

Тема: «Java. Монитор»

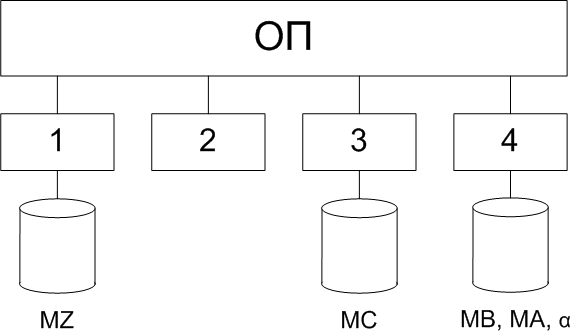
Выполнил: студент группы ИВ-83

НТУУ «КПИ» ФИВТ

Гульченко Владимир

**Техническое задание**

MA=MB + α \* (MС \* MZ)



**Этап 1. Построение параллельного алгоритма**

Вычисление данного матричного уравнения можно разбить на шаги:

1. MAH = MBH + α \* (MC \* MZH)

**Общие ресурсы:** α, MC

**Этап 2. Разработка алгоритмов процессов (задач)**

**№ Т1**

1. Ввод MZ
2. Cигнал задачам Т2, Т3, Т4 о завершении ввода
3. Ожидание ввода в задачах Т3, Т4
4. Копирование α1:=α, MС1:=MС
5. Счёт MAH = MBH + α1 \* (MC1 \* MZH)
6. Сигнал задаче Т4 о конце счёта

**№ Т2**

1. Ожидание ввода в задачах Т1, Т3, Т4
2. Копирование α2:=α, MС2:=MС
3. Счёт MAH = MBH + α2 \* (MC2 \* MZH)
4. Сигнал задаче Т4 о конце счёта

**№ Т3**

1. Ввод MС
2. Cигнал задачам Т1, Т2, Т4 о завершении ввода
3. Ожидание ввода в задачах Т1, Т4
4. Копирование α3:=α, MС3:=MС
5. Счёт MAH = MBH + α3 \* (MC3 \* MZH)
6. Сигнал задаче Т4 о конце счёта

**№ Т4**

1. Ввод MB, α
2. Cигнал задачам Т1, Т2, Т3 о завершении ввода
3. Ожидание ввода в задачах Т1, Т3
4. Копирование α4:=α, MС4:=MС
5. Счёт MAH = MBH + α4 \* (MC4 \* MZH)
6. Ожидание окончания счёта в Т1, Т2, Т3
7. Вывод МА

**Этап 3. Разработка схемы взаимодействия задач**



**Этап 4. Разработка программы**

**//МОНИТОР**

**public** **class** Data {

**private** **int** F1, F2;

**private** **int** [][] MC;

**private** **int** a;

**public** Data() {

F1 = 0;

F2 = 0;

}

//метод копирования Общих Ресурсов

**public** **int**[][] copyMC() {

**return** Lab5.*clone*(MC);

}

**public** **int** copya() {

**return** a;

}

//метод сихронизации по вводу

**public** **synchronized** **void** InputSynchro() {

**try** {

**while** (F2 < 3) {

wait();

}

} **catch** (InterruptedException e) {

e.printStackTrace();

}

}

//Сигнал об окончании ввода

**public** **synchronized** **void** InputSignal() {

F2++;

notifyAll();

}

//метод сихронизации по счёту (выводу)

**public** **synchronized** **void** OutputSynchro() {

**try** {

**while** (F1 < Lab5.*P* - 1) {

wait();

}

} **catch** (InterruptedException e) {

e.printStackTrace();

}

}

//Сигнал об окончании счёта

**public** **synchronized** **void** CalcEndSignal() {

F1++;

notifyAll();

}

**public** **synchronized** **void** writeMC(**int** [][] MT1) {

MC = Lab5.*clone*(MT1);

}

**public** **synchronized** **void** writea(**int** t) {

a = t;

}

}

/\*--------------------------------------------------------------

-- Paralel and distributed computing --

-- Laboratory work #5. Java. Monitors --

-- Func: MAH = MBH + alfa \* (MC \* MZH) --

-- IO-83 Gulchenko Volodumir --

--------------------------------------------------------------\*/

**public** **class** Lab5 {

**public** **static** **final** **int** *N* = 4;

**public** **static** **final** **int** *P* = 4;

**public** **static** **final** **int** *H* = *N*/*P*;

**static** **int** *MB*[][] = **new** **int**[*N*][*N*];

**static** **int** *MZ*[][] = **new** **int**[*N*][*N*];

**static** **int** *MA*[][] = **new** **int**[*N*][*N*];

**public** **static** **int**[][] clone(**int** [][] MM) {

**int** MT[][] = **new** **int** [MM.length][MM[0].length];

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

**for** (**int** j = 0; j < MM[0].length; j++) {

MT[i][j] = MM[i][j];

}

}

**return** MT;

}

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

Data dat = **new** Data();

T1 t1 = **new** T1(dat, 1);

T2 t2 = **new** T2(dat, 2);

T3 t3 = **new** T3(dat, 3);

T4 t4 = **new** T4(dat, 4);

t1.start();

t2.start();

t3.start();

t4.start();

System.*out*.println("КОНЕЦ");

}

}

**public** **class** T1 **extends** Tx {

**public** T1(Data d, **int** n) {

**super**(d, n);

}

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

System.*out*.println("Задача T" + num + " стартовала!");

//1. Ввод MZ

Lab5.*MZ* = input();

//2. Cигнал задачам Т2, Т3, Т4 о завершении ввода

dat.InputSignal();

//3. Ожидание ввода в задачах Т3, Т4

dat.InputSynchro();

//4. Копирование a1:=a, MС1:=MС

ax = dat.copya();

MCx = dat.copyMC();

//5. Счёт MAH = MBH + a1 \* (MC1 \* MZH)

process();

//6. Сигнал задаче Т4 о конце счёта

dat.CalcEndSignal();

System.*out*.println("Задача T" + num + " финишировала!");

}

}

**public** **class** T2 **extends** Tx {

**public** T2(Data d, **int** n) {

**super**(d, n);

}

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

System.*out*.println("Задача T" + num + " стартовала!");

//1. Ожидание ввода в задаче Т4

dat.InputSynchro();

//2. Копирование a2:=a, MZ2:=MZ, MR2:=MR

MCx = dat.copyMC();

ax = dat.copya();

//3. Счёт MAH = MBH + a2 \* (MC2 \* MZH)

process();

//4. Сигнал задаче Т4 о конце счёта

dat.CalcEndSignal();

System.*out*.println("Задача T" + num + " финишировала!");

}

}

**public** **class** T3 **extends** Tx {

**public** T3(Data d, **int** n) {

**super**(d, n);

}

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

System.*out*.println("Задача T" + num + " стартовала!");

//1. Ввод MС

MCx = input();

dat.writeMC(MCx);

//2. Cигнал задачам Т1, Т2, Т4 о завершении ввода

dat.InputSignal();

//3. Ожидание ввода в задачах Т1, Т4

dat.InputSynchro();

//4. Копирование a3:=a, MС3:=MС

MCx = dat.copyMC();

ax = dat.copya();

//5. Счёт MAH = MBH + a3 \* (MC3 \* MZH)

process();

//6. Сигнал задаче Т4 о конце счёта

dat.CalcEndSignal();

System.*out*.println("Задача T" + num + " финишировала!");

}

}

**public** **class** T4 **extends** Tx {

**public** T4(Data d, **int** n) {

**super**(d, n);

}

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

System.*out*.println("Задача T" + num + " стартовала!");

//1. Ввод MB, a

Lab5.*MB* = input();

ax = 1;

dat.writea(ax);

//2. Cигнал задачам Т1, Т2, Т3 о завершении ввода

dat.InputSignal();

//3. Ожидание ввода в задачах Т1, Т3

dat.InputSynchro();

//4. Копирование a4:=a, MС4:=MС

MCx = dat.copyMC();

ax = dat.copya();

//5. Счёт MAH = MBH + a4 \* (MC4 \* MZH)

process();

//6. Ожидание окончания счёта в Т1, Т2, Т3

dat.OutputSynchro();

//7. Вывод МА

**if** (Lab5.*N* <= 8) {

**for** (**int** i = 0; i < Lab5.*N*; i++) {

**for** (**int** j = 0; j < Lab5.*N*; j++) {

System.*out*.print(Lab5.*MA*[i][j] + " ");

}

System.*out*.println();

}

}

System.*out*.println("Задача T" + num + " финишировала!");

}

}