

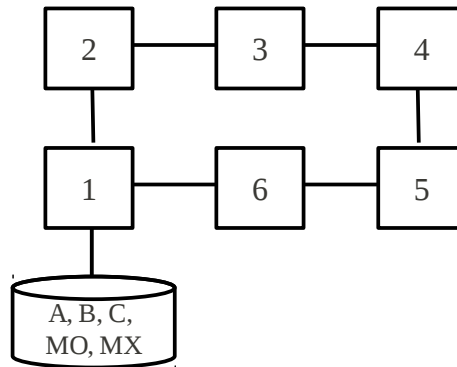
Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
Факультет інформатики та обчислюваної техніки
Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА №7
з курсу “Паралельні та розподілені обчислення”

Виконав: студент 3 курсу
ФІОТ групи ІО-72
Паскалов Петро
Перевірів: Корочкін О.В.

Київ, 2010

ТЗ на лабораторну роботу



$$A = B - C * (MO * MX)$$

Виконання лабораторної роботи

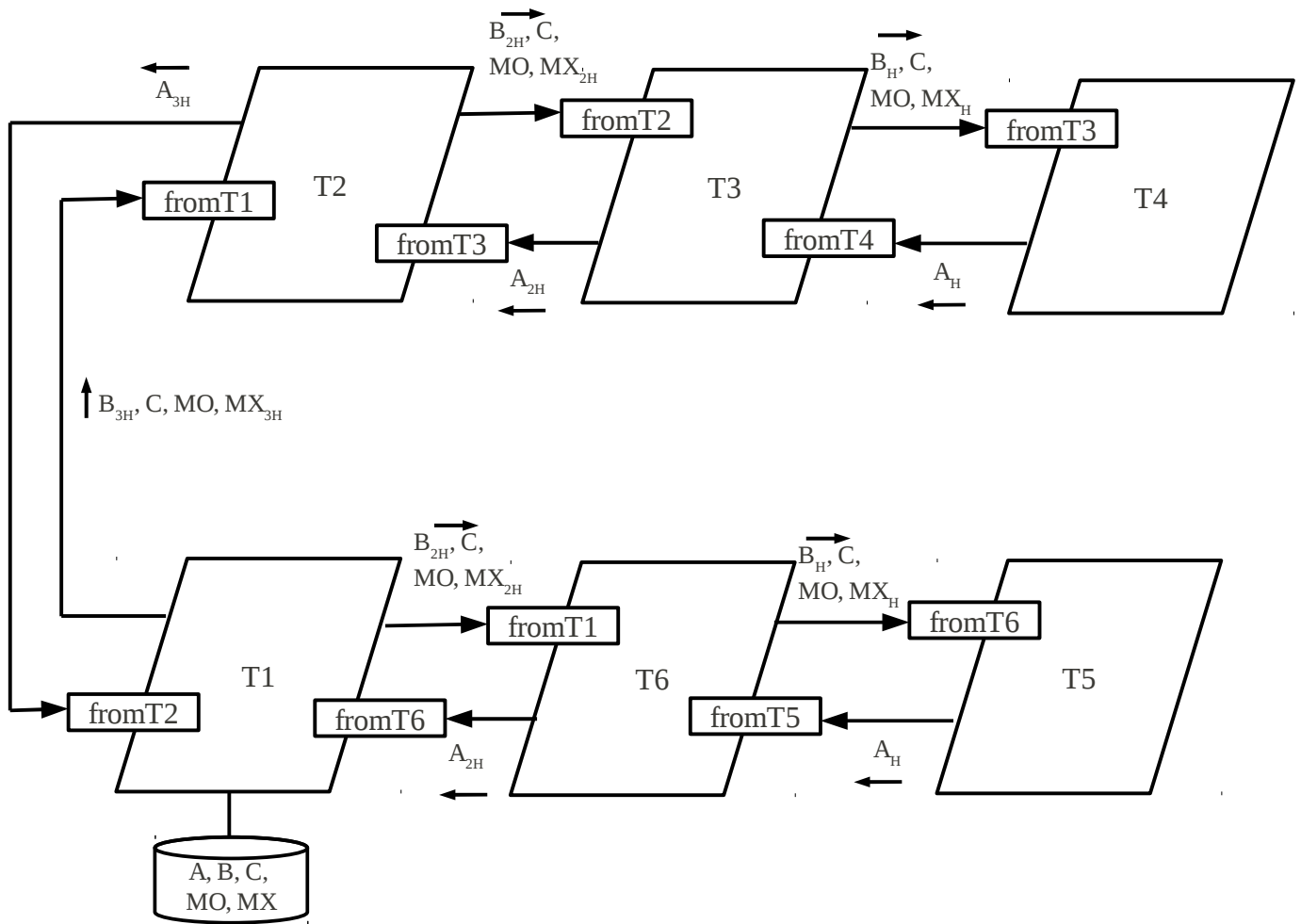
Етап 1. Розробка паралельного математичного алгоритму:

$$A_H = B_H - C * (MO * MX_H)$$

Етап 2. Розробка алгоритмів процесів (задач)

Т1	Т2	Т3
1. Ввід В, С, МО, МХ 2. Передати В _{ЗН} , С, МО, МХ _{ЗН} в Т2 3. Передати В _{2Н} , С, МО, МХ _{2Н} в Т6 4. Розрахунок А _Н = В _Н - С*(МО*МХ _Н) 5. Прийняти А _{2Н} від Т6 6. Прийняти А _{3Н} від Т2 7. Вивід А	1. Прийняти В _{ЗН} , С, МО, МХ _{ЗН} від Т1 2. Передати В _{2Н} , С, МО, МХ _{2Н} в Т3 3. Розрахунок А _Н = В _Н - С*(МО*МХ _Н) 4. Прийняти А _{2Н} від Т3 5. Передати А _{3Н} в Т1	1. Прийняти В _{2Н} , С, МО, МХ _{2Н} від Т2 2. Передати В _Н , С, МО, МХ _Н в Т4 3. Розрахунок А _Н = В _Н - С*(МО*МХ _Н) 4. Прийняти А _Н від Т4 5. Передати А _{2Н} в Т2
Т4	Т5	Т6
1. Прийняти В _Н , С, МО, МХ _Н від Т3 2. Розрахунок А _Н = В _Н - С*(МО*МХ _Н) 3. Передати А _Н в Т3	1. Прийняти В _Н , С, МО, МХ _Н від Т6 2. Розрахунок А _Н = В _Н - С*(МО*МХ _Н) 3. Передати А _Н в Т6	1. Прийняти В _{2Н} , С, МО, МХ _{2Н} від Т1 2. Передати В _Н , С, МО, МХ _Н в Т5 3. Розрахунок А _Н = В _Н - С*(МО*МХ _Н) 4. Прийняти А _Н від Т5 5. Передати А _{2Н} в Т1

Етап 3. Розробка схеми взаємодії задач



Етап 4. Розробка програми

```
-----
-- PRO                                     --
-- Lab7.Ada.Randevu                       --
-- A = B - C*(MO*MX);                     --
-- Paskalov Petr, IO-72                   --
-- 03.06.10                               --
-----

with Ada.Text_IO, Ada.Integer_Text_IO;
use Ada.Text_IO, Ada.Integer_Text_IO;

procedure laba7 is

    N: integer := 12;
    P: integer := 6;
    H: integer := N/P;

    type Vector_Base is array (Integer range <>) of Integer;
    subtype Vector is Vector_Base (1..N);
    subtype VectorH is Vector_Base (1..H);
    subtype Vector2H is Vector_Base (1..2*H);
    subtype Vector3H is Vector_Base (1..3*H);

    type Matrix_Base is array (Integer range <>) of Vector;
    subtype Matrix is Matrix_Base(1..N);
    subtype MatrixH is Matrix_Base(1..H);
    subtype Matrix2H is Matrix_Base(1..2*H);
    subtype Matrix3H is Matrix_Base(1..3*H);

    procedure calculation(A_h: out VectorH; B_h: in VectorH; C: in Vector;
MX_h: in MatrixH; MO: in Matrix) is
        temp: integer;

    begin

        for i in 1..H loop
            A_h(i):=0;
            for j in 1..N loop
                temp:=0;
                for r in 1..N loop
                    temp:= temp + MO(r)(j)*MX_h(i)(r);
                end loop;
                A_h(i) := A_h(i) + C(j)*temp;
            end loop;
            A_h(i) := B_h(i) - A_h(i);
        end loop;

    end calculation;

    procedure Tasks is
    task T1 is
        entry fromT2(A_3h: in Vector3H);
        entry fromT6(A_2h: in Vector2H);
    end T1;

    task T2 is
        entry fromT1(MX_3h: in Matrix3H; tC: in Vector; tMO: in Matrix;
B_3h: in Vector3H);
        entry fromT3(A_2h: in Vector2H);
    end T2;

    task T3 is
        entry fromT2(MX_2h: in Matrix2H; tC: in Vector; tMO: in Matrix;
```

```

B_2h: in Vector2H);
    entry fromT4(A_h: in VectorH);
end T3;

task T4 is
    entry fromT3(MX_h: in MatrixH; tC: in Vector; tMO: in Matrix; B_h:
in VectorH);
end T4;

task T5 is
    entry fromT6(MX_h: in MatrixH; tC: in Vector; tMO: in Matrix; B_h:
in VectorH);
end T5;

task T6 is
    entry fromT1(MX_2h: in Matrix2H; tC: in Vector; tMO: in Matrix;
B_2h: in Vector2H);
    entry fromT5(A_h: in VectorH);
end T6;

```

```

-----Task 1-----
task body T1 is
    MO, MX: Matrix;
    A, B, C: Vector;
begin
    Put_line("T1 is started. ");
    for i in 1..N loop
        B(i):=1;
        C(i):=1;
        for j in 1..N loop
            MO(i)(j):=1;
            MX(i)(j):=1;
        end loop;
    end loop;

    T2.fromT1(MX(1..3*H), C, MO, B(1..3*H));
    T6.fromT1(MX(3*H+1..5*H), C, MO, B(3*H+1..5*H));

    calculation(A(1..H), B(5*H+1..N), C, MX(5*H+1..N), MO);

    accept fromT2(A_3h: in Vector3H) do
        A(H+1..4*H):=A_3h;
    end fromT2;

    accept fromT6(A_2h: in Vector2H) do
        A(4*H+1..N):=A_2h;
    end fromT6;

    if N<3 then
        for i in 1..N loop
            Put(A(i)(j));
        end loop;
        New_Line;
    end if;

    Put_line("T1 is finished. ");
end T1;

```

```

-----Task 2-----
task body T2 is
    MO: Matrix;
    MX3h: Matrix3H;
    B3h, A3h,: Vector3H;
    C: Vector;

begin
    Put_line("T2 is started. ");

```

```

        accept fromT1(MX_3h: in Matrix3H; tC: in Vector; tMO: in Matrix;
B_3h: in Vector3H) do
            B3h:= B_3h;
            MX3h:=      MX_3h;
            MO:=tMO;
            C:=tC;
        end fromT1;

        T3.fromT2(MX3h(1..2*H), C, MO, B3h(1..2*H));

        calculation(A3h(1..H), B3h(2*H+1..3*H), C, MX3h(2*H+1..3*H), MO);

        accept fromT3(A_2h: in Vector2H) do
            A3h(H+1..2*H):=A_2h;
        end fromT3;

        T1.fromT2(A3h);
        Put_line("T2 is finished. ");
    end T2;

```

-----Task 3-----

```

    task body T3 is
        MO: Matrix;
        MX2h: Matrix2H;
        B2h, A2h,: Vector2H;
        C: Vector;

    begin
        Put_line("T2 is started. ");

        accept fromT2(MX_2h: in Matrix2H; tC: in Vector; tMO: in Matrix;
B_2h: in Vector2H) do
            B2h:= B_2h;
            MX2h:=      MX_2h;
            MO:=tMO;
            C:=tC;
        end fromT2;

        T4.fromT3(MX2h(1..H), C, MO, B2h(1..H));

        calculation(A2h(1..H), B2h(H+1..2*H), C, MX3h(H+1..2*H), MO);

        accept fromT4(A_h: in VectorH) do
            A2h(H+1..2*H):=A_h;
        end fromT4;

        T2.fromT3(A2h);
        Put_line("T3 is finished. ");
    end T3;

```

-----Task 4-----

```

    task body T4 is
        MO: Matrix;
        MXh: MatrixH;
        Bh, Ah,: VectorH;
        C: Vector;

    begin
        Put_line("T4 is started. ");

        accept fromT3(MX_h: in MatrixH; tC: in Vector; tMO: in Matrix; B_h:
in VectorH) do
            Bh:= B_h;
            MXh:= MX_h;
            MO:=tMO;
            C:=tC;
        end fromT3;

```

```

        calculation(Ah, Bh), C, MXh, MO);
        T3.fromT4(Ah);

        Put_line("T4 is finished. ");
    end T4;

-----Task 4-----
    task body T4 is
        MO: Matrix;
        MXh: MatrixH;
        Bh, Ah,: VectorH;
        C: Vector;

    begin
        Put_line("T5 is started. ");

        accept fromT6(MX_h: in MatrixH; tC: in Vector; tMO: in Matrix; B_h:
in VectorH) do
            Bh:= B_h;
            MXh:= MX_h;
            MO:=tMO;
            C:=tC;
        end fromT6;

        calculation(Ah, Bh), C, MXh, MO);
        T6.fromT5(Ah);

        Put_line("T5 is finished. ");
    end T4;

-----Task 6-----
    task body T6 is
        MO: Matrix;
        MX2h: Matrix2H;
        B2h, A2h,: Vector2H;
        C: Vector;

    begin
        Put_line("T6 is started. ");

        accept fromT1(MX_2h: in Matrix2H; tC: in Vector; tMO: in Matrix;
B_2h: in Vector2H) do
            B2h:= B_2h;
            MX2h:= MX_2h;
            MO:=tMO;
            C:=tC;
        end fromT1;

        T5.fromT6(MX2h(1..H), C, MO, B2h(1..H));

        calculation(A2h(1..H), B2h(H+1..2*H), C, MX3h(H+1..2*H), MO);

        accept fromT5(A_h: in VectorH) do
            A2h(H+1..2*H):=A_h;
        end fromT5;

        T1.fromT6(A2h);
        Put_line("T6 is finished. ");
    end T6;

begin
    null;
end Tasks;

begin
    put_line("Main is started. ");
    Tasks;
    Put_line("Main is finished. ");
end laba7;

```