

Національний технічний університет України

«Київський Політехнічний Інститут»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №1

З предмету «Паралельні та розподілені обчислення»

Ада. Семафори

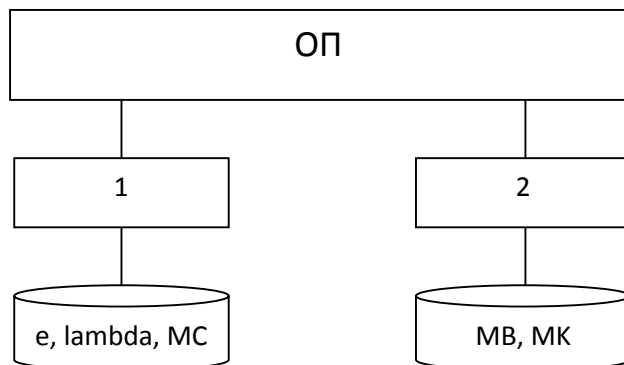
Виконав:

Студент
III курсу ФІОТ
групи ІО-12
Бута С. О.

Залікова книжка
№1205

Техническое задание

1. Структура ПКС с ОП:



2. Задача: $e = \max(MB + \lambda * MC * MK)$.
3. Язык программирования: Ада.
4. Средства взаимодействия задач: семафоры.

Выполнение работы

1. Разработка параллельного математического алгоритма.
 - 1) $e_i = \max(MB_H + \lambda * MC_H * MK)$; ОП: λ, MK .
 - 2) $e = \max(e, e_i)$; ОП: e .

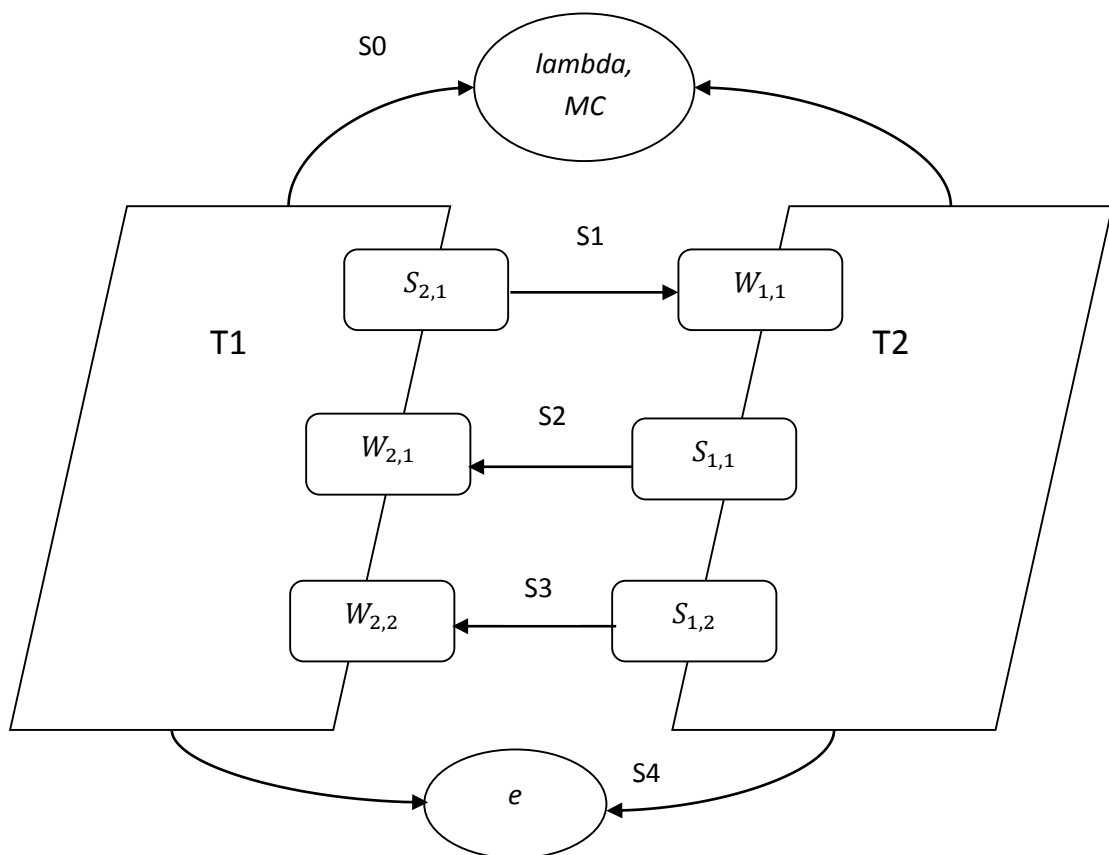
2. Разработка алгоритмов процессов.

Задача Т1:

- 1) Ввод λ, MC .
- 2) Сигнал Т2 о завершении ввода. $S_{2,1}$
- 3) Ждать завершения ввода в Т2. $W_{2,1}$
- 4) Копии КУ
 - $\lambda_1 = \lambda$
 - $MK_1 = MK$
- 5) Счет1
$$e_1 = \max(MB_H + \lambda_1 * MC_H * MK_1)$$
- 6) Счет2 КУ
$$e = \max(e, e_1)$$
- 7) Ждать завершения счета в Т2. $W_{2,2}$
- 8) Вывод e .

Задача Т2:

- 1) Ввод MB, MK .
- 2) Ждать завершения ввода в Т1. $W_{1,1}$
- 3) Сигнал Т1 о завершении ввода. $S_{1,1}$
- 4) Копии KY
 - $lambda2 = lambda$
 - $MK2 = MK$
- 5) Счет1 $e2 = max(MB_H + lambda2 * MC_H * MK2)$
- 6) Счет2 KY
 $e = max(e, e2)$
- 7) Сигнал Т1 о завершении счета. $S_{1,2}$

Разработка схемы взаимодействия процессов:**Разработка программы.****Листинг:**

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

```
-----
-- Лабораторна робота №1. Ада. Семафори
-- Бута С. О.
-- Математична операція: e=max(MB+lambda*MC*MK)
-- Дата: 13.03.14
-----
```

```
procedure Lab1 is
```

```

N: integer := 400;
P: integer := 2;
H: integer := N/P;
filler: integer := 1;
lambda: integer;
e: integer := Integer'First;

```

```

type Vector is array(1..N) of integer;
type Matrix is array(1..N) of Vector;

```

```

MB, MC, MK: Matrix;
Sem0, Sem1, Sem2, Sem3, Sem4: Suspension_Object;

```

```

procedure MatrixInput(m : out Matrix) is
begin
  for i in 1..N loop
    for j in 1..N loop
      m(i)(j) := filler;
    end loop;
  end loop;
end MatrixInput;

```

```

procedure MatrixOutput(m : in Matrix) is
begin
  for i in 1..N loop
    for j in 1..N loop
      put(m(i)(j));
      put(" ");
    end loop;
    Put_Line("");
  end loop;
end MatrixOutput;

```

```

procedure MatrixCopy (mOut : out Matrix; mIn : in Matrix) is
begin
  for i in 1..N loop
    for j in 1..N loop
      mOut(i)(j) := mIn(i)(j);
    end loop;
  end loop;
end MatrixCopy;

```

```

--запуск задач
procedure StartTasks is
--задача 1
task T1 is
  pragma Storage_Size(500_000_000);
end T1;
task body T1 is
  lambda1, e1, a1: integer;
  MK1: Matrix;
begin
  Put_Line("Task T1 is started.");
  --Ввод lambda, MC
  lambda := filler;
  MatrixInput(MC);
  --Сигнал T2 о завершении ввода
  Set_True(Sem1);
  --Ждать завершения ввода в T2.
  Suspend_Until_True(Sem2);
  --Копии
  Suspend_Until_True(Sem0);
  MatrixCopy(MK1, MK);
  lambda1 := lambda;
  Set_True(Sem0);

```

```

e1 := Integer'First;
--Счет1
for i in 1..H loop
  for j in 1..N loop
    a1 := 0;
    for k in 1..N loop
      a1 := a1+MC(i)(k)*MK1(k)(j);
    end loop;
    a1 := MB(i)(j) + lambda1*a1;
    if e1<a1 then
      e1 := a1;
    end if;
  end loop;
end loop;

--Счет2
Suspend_Until_True(Sem4);
if e<e1 then
  e := e1;
end if;
Set_True(Sem4);
--Ждать завершения счета в T2
Suspend_Until_True(Sem3);
--Вывод e.
Put("Result: e =");
Put(e, 6);
Put_line(" ");
Put_Line("Task T1 is finished.");
end T1;

```

```

--задача 2
task T2 is
  pragma Storage_Size(500_000_000);
end T2;
task body T2 is
  lambda2, e2, a2: integer;
  MK2: Matrix;
begin
  Put_Line("Task T2 is started.");
  --Ввод MB, MK
  MatrixInput(MB);
  MatrixInput(MK);
  --Ждать завершения ввода в T1
  Suspend_Until_True(Sem1);
  --Сигнал T1 о завершении ввода.
  Set_True(Sem2);
  --Копии
  Suspend_Until_True(Sem0);
  MatrixCopy(MK2, MK);
  lambda2 := lambda;
  Set_True(Sem0);

```

```

e2 := Integer'First;
--Счет1
for i in H..N loop
  for j in 1..N loop
    a2 := 0;
    for k in 1..N loop
      a2 := a2+MC(i)(k)*MK2(k)(j);
    end loop;
    a2 := MB(i)(j) + lambda2*a2;
    if e2<a2 then
      e2 := a2;
    end if;

```

```

        end loop;
    end loop;

    --Счет2
    Suspend_Until_True(Sem4);
    if e<e2 then
        e := e2;
    end if;
    Set_True(Sem4);
    --Сигнал T1 о завершении счета.
    Set_True(Sem3);
    Put_Line("Task T2 is finished.");
end T2;

begin
    put("");
end StartTasks;

begin
    put_line("Main procedure started");

    Set_True(Sem0);
    Set_True(Sem4);

    StartTasks;

    put_line("Main procedure finished");
end Lab1;

```