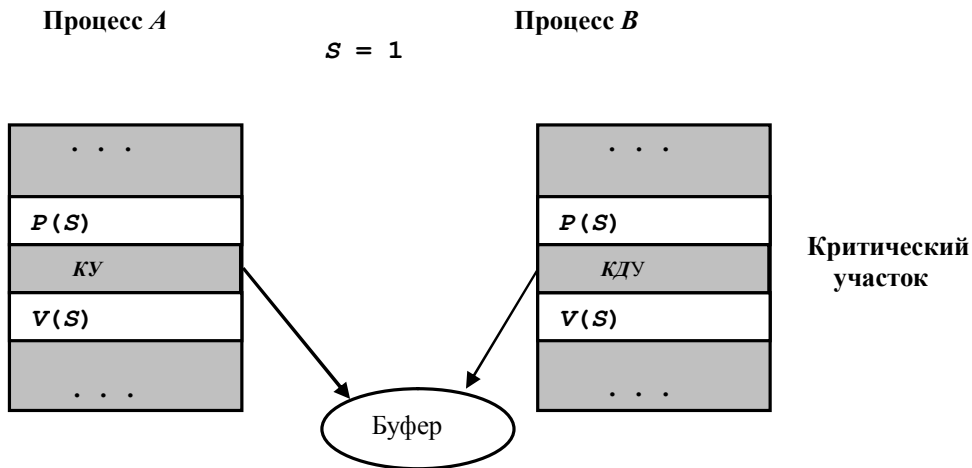


Завдання № 1

Розробити структурну схему вирішення завдання взаємного відключення двох процесів при роботі зі спільним ресурсом Буфер. Застосувати бінарні семафори та операції $P(S)$ і $V(S)$. Навести алгоритми виконання операцій $P(S)$ і $V(S)$.

Ответ:



Операция $P(S)$:

1. Проверить значение S .
2. Если $S = 0$, то заблокировать процесс
3. Иначе $S := S - 1$

Операция $V(S)$:

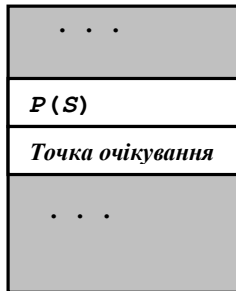
1. $S := S + 1$

Завдання № 2

Розробити структурну схему вирішення завдання синхронізації двох задач . Застосувати бінарні семафори та операції $P(S)$ і $V(S)$.

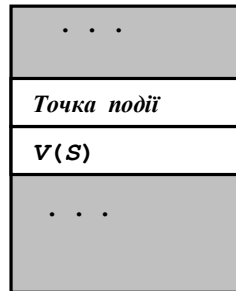
Відповідь :

Процес *A*



$S = 0$

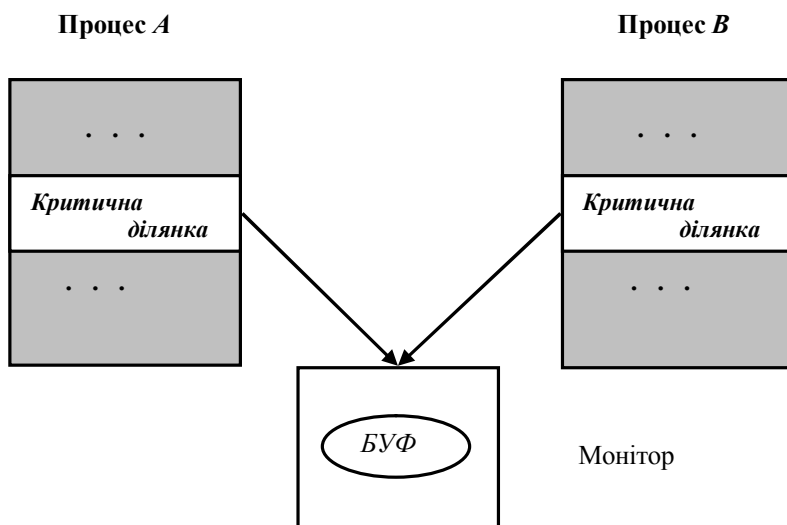
Процес *B*



Завдання № 3

Розробити структурну схему вирішення завдання взаємного відключення двох задач при роботі зі спільним ресурсом БУФ. Застосувати механізм моніторів. Визначити властивості процедур монітора.

Відповідь :

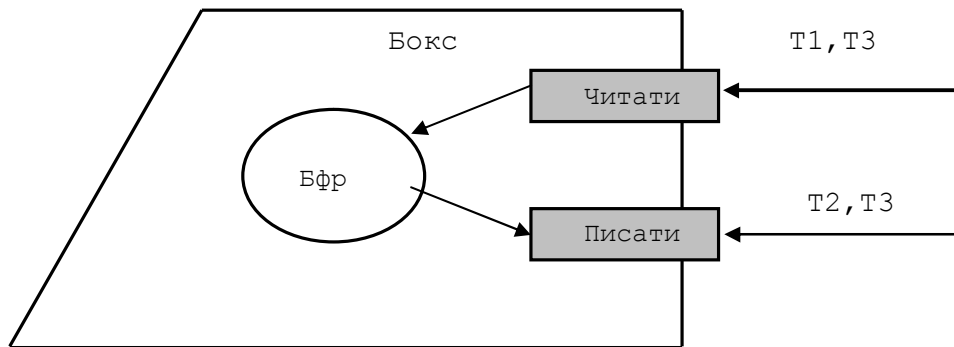


Властивість процедур монітора – вони виконуються в режимі взаємного виключення, тобто монітор дозволяє виконання тільки однієї своєї процедури.

Завдання № 4

Розробити структурну схему захищеного модулю Бокс для вирішення завдання взаємного відключення трьох задач T1, T2, T3 при роботі зі спільним ресурсом Бфр. Реалізувати за допомогою мови Ада.

Відповідь :



```
-- захищений модуль
protected Бокс is
  procedure Читати(X: out integer);
  procedure Писати(x: in integer);
private
  Бфр : integer:= 10;      -- спільний ресурс
end Бокс;

-- тіло захищеного модуля
protected body Бокс is

  procedure Читати(X: out integer) is
  begin
    X := Бфр;
  end;

  procedure Писати(x: in integer) is
  begin
    Бфр := X;
  end ;

end Бокс;
```

Завдання № 5

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі А і В . Встановити пріоритети задач: 2 – для задачі А, 5 – для задачі В . Використати оператор `delay` для затримки задачі А на 4,5 сек , задачі В - на 7,4 сек. , основної програми – на 12,25 сек.

Відповідь :

```
procedure Lab is

    task A is
        pragma Priority(2);
    end A;

    task B is
        pragma Priority(5);
    end B;

    task body A is
    begin
        delay(4.5);
    end A;
    -----
    task body B is
    begin
        delay(7.4);
    end B;

begin

    delay(12.25);

end Lab;
```

Завдання № 6

Розробити програму мовою Ада, в якій створюють та запускають на виконання п'ять задач T1 .. T5. Використати задачний тип, в якому за допомогою дискримінанту визначити для кожної задачі внутрішній ідентифікатор (Tid).

Відповідь :

```
procedure Lab is

    -- задачний тип з дискримінантом
    task type ЗадачнТип (Номер : integer);

    task body ЗадачнТип is
        Tid : integer := Номер;
    begin
        put (Номер_Задачі);
    end ЗадачнТип;

    T1: ЗадачнТип(1);
    T2: ЗадачнТип(2);
    T3: ЗадачнТип(3);
    T4: ЗадачнТип(4);
    T5: ЗадачнТип(5);

begin
    null;
end Lab;
```

Завдання № 7

Розробити програму мовою Java , в якій створюються та запускаються на виконання два потоки А, В . Встановити пріоритети потоків: 2 – для задачі А, 5 – для задачі В . Використати оператор `sleep` для затримки задачі А на 4,5 сек , задачі В - на 7,4 сек.

Відповідь :

```
class AT extends Thread {

    // перевизначення методу run()
    public void run(){
        sleep(4500);
    }
} // AT

class BT extends Thread {

    public void run(){
        sleep(7400);
    }
} // BT

// головний потік
class Lab {

    // точка входу в основний клас
    public static void main(String args []){

        // оголошення екземплярів потоків
        AT A = new AT();
        BT B = new BT();

        // пріоритети потоків
        A.Set_Priority(2);
        B.Set_Priority(5);

        // запуск потоків
        A.start();
        B.start();

    } // main
} // Lab
```

Завдання № 8

Розробити програму мовою Java , в якій створюють та запускають на виконання два потоку А і В . Потоки працюють зі спільним ресурсом Буфер . Розробити клас , який містить спільний ресурс і забезпечує синхронізований доступ потоків А і В до нього для читання та запису даних.

Відповідь :

```
class Доступ{

    private int Буфер;           // спільний ресурс

    synchronized int Читати() {
        return Буфер;
    }
    synchronized void Писати(int x) {
        Буфер = x;
    }
} // Доступ

public class aThread extends Thread{

    int a1;
    Доступ Z;

    public aThread(Доступ q) {
        Z = q;
    }

    public void run() {
        a1= Z.Читати();
    }

} // А

public class bThread extends Thread{

    int a2;
    Доступ Z;

    public bThread(Доступ q) {
        Z = q;
    }

    public void run() {
        Z.Писати(a2);
    }

} // В
```


Завдання № 9

Розробити програму мовою Java , в якій створюють та запускають на виконання два потоки А і В. Виконати вирішення завдання синхронізації потоків через додатковий клас – Control. Потік А чекає на подію в потоці В. Навести структуру класу Control.

Відповідь :

```
public class Control {

    private int Fl = 0;
    public synchronized void Чекати() {
        try{
            if (Fl == 0)
                wait();
        }catch(Exception e) {}
    }
    public synchronized
        void Сигнал() {
        notify();
        Fl = 1;
    }
} // Control

public class aThread extends Thread{
    Data Z;
    public aThread(Data q){
        Z = q;
    }
    public void run(){
        //Очікування на подію в потоці В
        Z.Чекати();
    }
}

public class bThread extends Thread{
    Data Z;
    public bThread(Data q) {
        Z = q;
    }
    public void run(){
        // сигнал про подію
        Z.Сигнал();
    }
}

// Основний клас
public class Lab {
    public static void main (String [] args){
        Data D = new Data();
        aThread A = new aThread(D);
        bThread B = new bThread(D);
        A.start();
        B.start();
    }
} //Lab
```

Завдання № 10

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В матрицю МТ. Навести структурну схему взаємодії задач.

Відповідь :

```
procedure Lab is

  type Matrix is array (1..5, 1 .. 5) of integer;

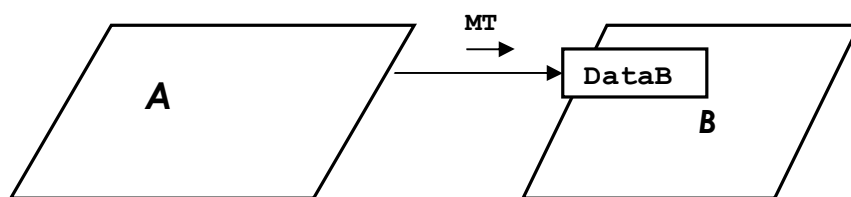
  task B is
    entry DataB(z: in Matrix);
  end B;
  task body B is
    MB: Matrix;
  begin

    accept DataB(z: in Matrix) is
      MB:= z;
    end Дані;

  end B;

  task A;
  task body B is
    MT : Matrix;
  begin
    B.DataB(MT);
  end A;

begin
  null;
end Lab;
```



Завдання № 11

Розробити програму мовою Ада, в якій створюються та запускаються на виконання чотири задачі A, B, C, D. За допомогою механізму рандеву передати зі задачі A в задачі B, C, D вектора VB, VC, VD. Навести структурну схему взаємодії задач.

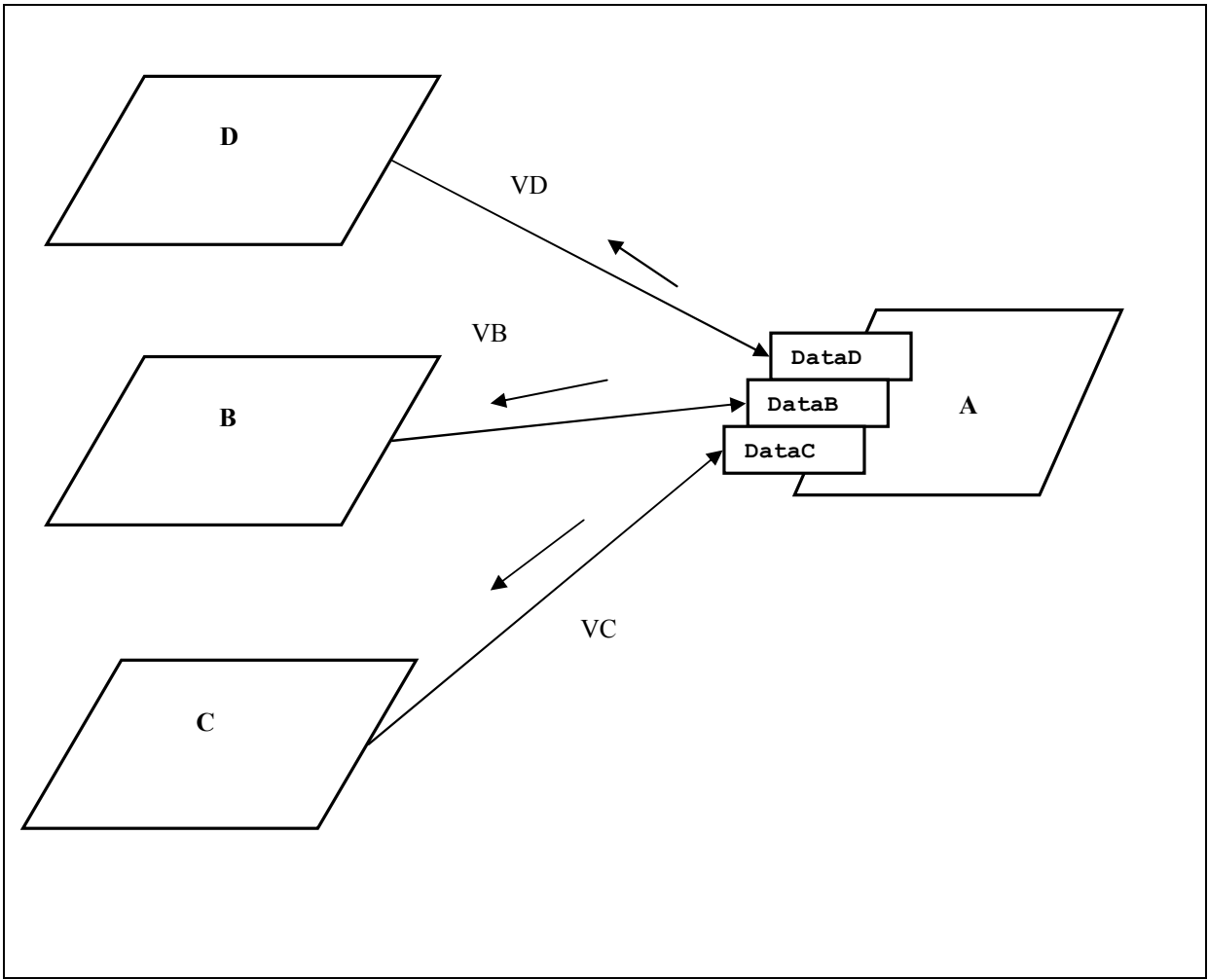
Відповідь :

```
procedure Lab is
  type Vector is array (1..10) of integer;
  task A is
    entry DataB(z: out Vector);
    entry DataC(z: out Vector);
    entry DataD(z: out Vector);
  end A;
  task B;
  task C;
  task D;

  task body A is
    VB, VC, VD: Vector;
  begin
    accept DataB(z: out Vector) is
      z:= VB;
    end DataB;
    accept DataC(z: out Vector) is
      z:= VB;
    end DataC;
    accept DataD(z: out Vector) is
      z:= VB;
    end DataD;
  end A;
  task body B is
    TB : Vector;
  begin
    A.DataB(TB);
  end B;

  task body C is
    TB : Vector;
  begin
    A.DataC(TB);
  end C;

  task body D is
    TB : Vector;
  begin
    A.DataD(TB);
  end D;
begin
  null;
end Lab;
```



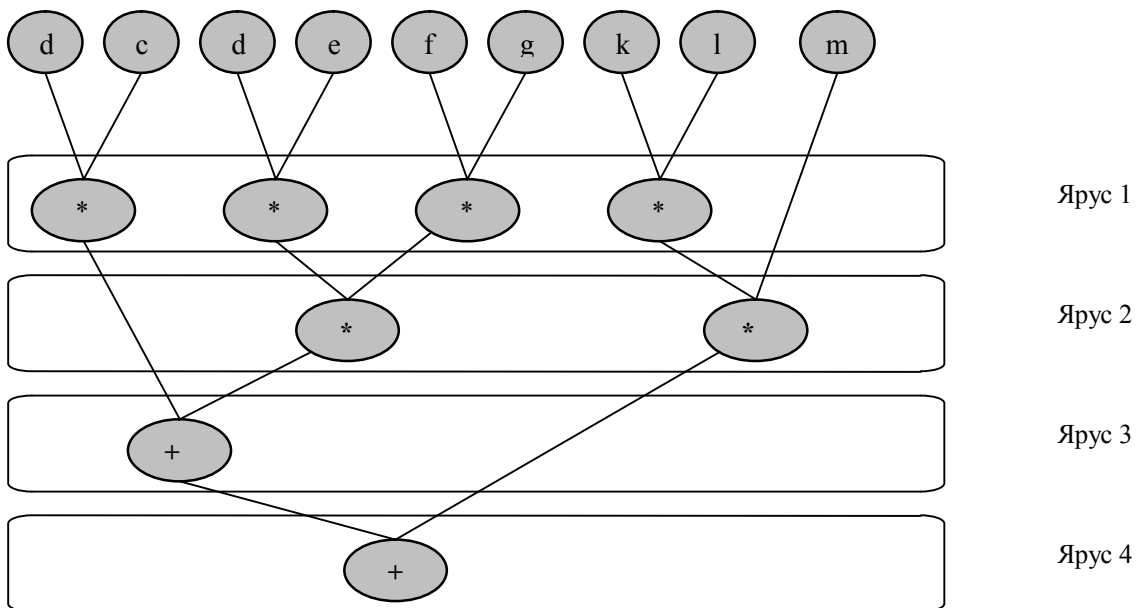
Завдання № 12

Виконати побудову паралельного алгоритму обчислення арифметичного виразу

$$a = b * c + d * e * f * g + k * l * m.$$

Побудувати ярусно-паралельну форму паралельного алгоритму, визначити її параметри (висоту H , ширину кожного ярусу - R_i , ширину всієї форми - R). Обчислити час виконання виразу в послідовній і паралельній системах, коефіцієнт прискорення K_p та коефіцієнт завантаження K_z .

Відповідь :



$$R_1 = 4 \quad R_2 = 2 \quad R_3 = 1 \quad R_4 = 1$$

$$R = \max \{R_1, R_2, R_3, R_4\} = 4 \quad P = 4$$

$$H = 4$$

$$T_1 = 9 \quad T_p = 4$$

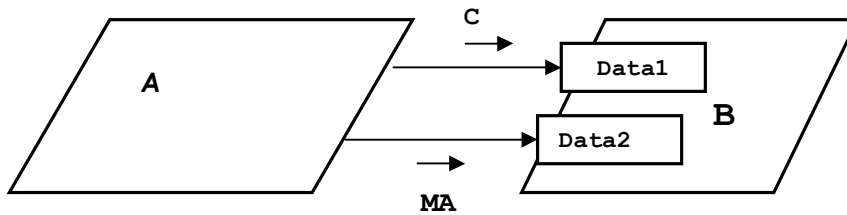
$$K_p = T_1 / T_p = 9 / 4 = 2,25$$

$$K_z = K_p / P = 2,25 / 4 = 0,56 \text{ (56\%)}$$

Завдання № 13

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В вектор С і потім - матрицю МА. Навести структурну схему взаємодії задач.

Відповідь :



```
procedure Lab is

    type Vector is array (1..10) of integer;
    type Matrix is array (1..10,1.. 5) of integer;

    task B is
        entry Data1(z: in Vector);
        entry Data2(z: in Matrix);
    end B;

    task body B is
        MT: Matrix;
        T: Vector;
    begin

        accept Data1(z: in Vector) is
            T:= z;
        end Data1;

        accept Data2(z: in Matrix) is
            MT:= z;
        end Data2;

    end B;

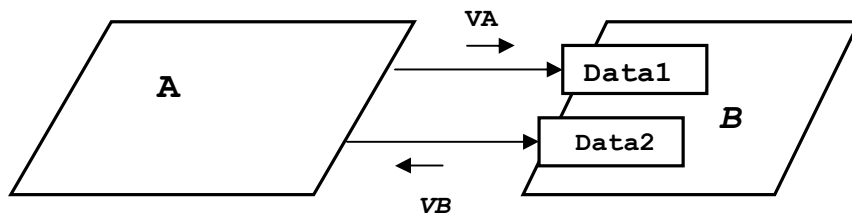
    task A;
    task body A is
        MA: Matrix;
        C: Vector;
    begin
        B.Data1(C);
        B.Data2(MA);
    end A;

begin
    null;
end Lab;
```

Завдання № 14

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі A і B. За допомогою механізму рандеву передати зі задачі A в задачу B вектор VA і прийняти вектор VB. Навести структурну схему взаємодії задач.

Відповідь :



```
procedure Lab is

    type Vector is array (1..10) of integer;

    task B is
        entry Data1(z: in Vector);
        entry Data2(z: out Vector);
    end B;

    task body B is
        VT1,VT2: Vector;
    begin

        accept Data1(z: in Vector) is
            VT1:= z;
        end Data1;

        accept Data2(z: out Vector) is
            z:= VT2;
        end Data2;

    end B;

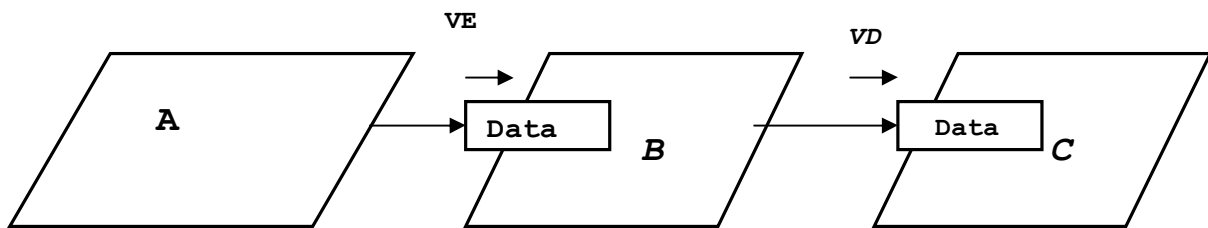
    task A;
    task body A is
        VA, VB : Vector;
    begin
        B.Data1(VA);
        B.Data2(VB);
    end A;

begin
    null;
end Lab;
```

Завдання № 15

Розробити програму мовою Ада, в якій створюють та запускають на виконання три задачі А, В і С. За допомогою механізму рандеву передати зі задачі А в задачу В вектор VE , а зі задачі В у задачу С передати вектор VD . Навести структурну схему взаємодії задач.

Відповідь :



```
procedure Lab is
  type Vector is array (1..4) of integer;
  task B is
    entry Data(z: in Vector);
  end B;
  task C is
    entry Data(z: in Vector);
  end C;

  task body B is
    VE, D: Vector;
  begin
    accept Data(z: in Vector) is
      VE := z;
    end Data1;
    B.Data2(E1);
  end B;

  task body C is
    VD: Vector;
  begin
    accept Data(z: in Vector) is
      VD := z;
    end Data1;
  end C;

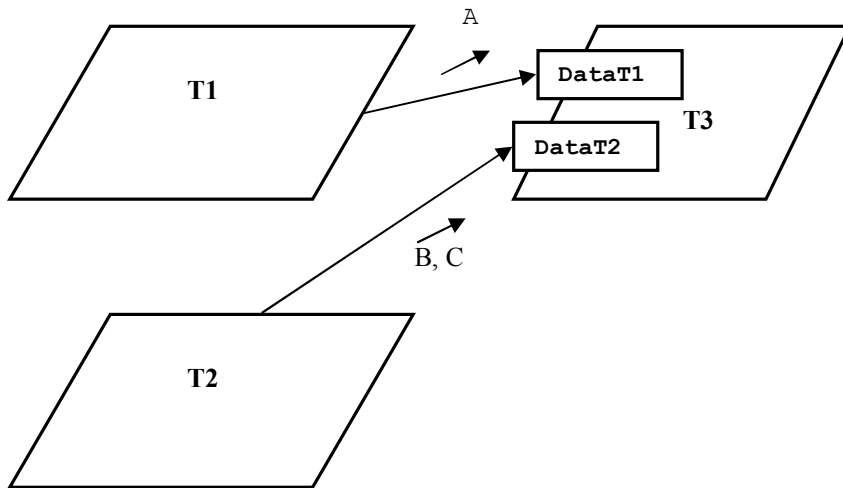
  task A;
  task body A is
    VE : Vector;
  begin
    B.Data(VE);
  end A;

begin
  null;
end Lab;
```


Завдання № 16

Розробити програму мовою Ада, в якій створюють та запускають на виконання три задачі T1, T2 і T3. За допомогою механізму рандеву передати зі задачі T1 в задачу T3 вектор A, а зі задачі T2 в задачу T3 передати два вектори B і C. Навести структурну схему взаємодії

Відповідь :



```
procedure Lab is
  N: integer:= 10;
  type Бектор is array(1..N) of integer;
  task T1;
  task T2;
  task T3 is
    entry DataT1(V:      in Бектор);
    entry DataT2(V1, V2: in Бектор);
  end T3;
  task body T1 is
    A: Бектор;
  begin
    T3.DataT1(A);
  end T1;
  task body T2 is
    B,C: Бектор;
  begin
    T3.DataT2(B,C);
  end T2;
  task body T3 is
    VA, VB, VC: Бектор;
  Begin
    accept DataT1(V: in Бектор) do
      VA:= V;
    end Data;
    accept DataT2(V1, V2: in Бектор) do
      VB:= V1;
      VC:= V2;
    end Data;
  end T3;
begin
  null;
end Lab;
```

Завдання № 17

Розробити програму мовою Java , в якій створюються та запускаються на виконання два потоки А і В. Виконати вирішення завдання взаємного виключення для потоків через додатковий клас - **Monitor**, якщо потоки працюють зі спільним ресурсом **Box** .
Навести структуру класу **Монітор** .

Відповідь :

```
public class Monitor {

    int Box = 0;          // спільний ресурс

    public synchronized void
        Читати() { return Box; }

    public synchronized
        Писати(int x) { Box = x; }

}

public class aThread extends Thread{
    int a1;
    Monitor Z;

    public aThread(Monitor q) {
        Z = q;
    }
    public void run() {
        a1= Z.Читати();
    }

} // А

public class bThread extends Thread{
    int a2;
    Monitor Z;
    public bThread(Monitor q) {
        Z = q;
    }
    public void run() {
        Z.Писати(a2);
    }
} // В

// Основний клас
public class Lab {
    public static void main (String [] args){
        Monitor D = new Monitor();

        aThread A = new aThread(D);
        bThread B = new bThread(D);
        A.start();
        B.start();
    }
} //Lab
```

Завдання № 18

Розробити програму мовою Ада, в якій створюють та запускають на виконання три задачі А, В і С. Встановити пріоритети задач: 3 – для задачі А, 5 – для задачі В, 7 – для задачі С. Використати оператор `delay` для затримки задачі А на 6 сек, задачі В - на 8 сек., С – на 11 сек.

Відповідь : `procedure Lab is`

```
task A is
  pragma Priority(3);
end A;
```

```
task B is
  pragma Priority(5);
end B;
```

```
task C is
  pragma Priority(7);
end C;
```

```
task body A is
begin
  delay(6);
end A;
```

```
-----
task body B is
begin
  delay(8);
end B;
```

```
-----
task body C is
begin
  delay(11);
end C;
```

```
begin
```

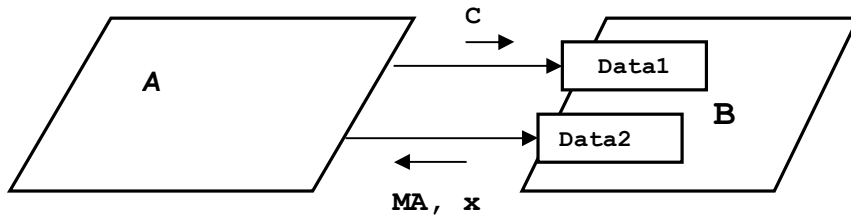
```
  null;
```

```
end Lab;
```

Завдання № 19

Розробити програму мовою Ада, в якій створюються та запускаються на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В вектор С і потім - прийняти матрицю МА та скаляр х. Навести структурну схему взаємодії задач.

Відповідь :



```
procedure Lab is

  type Vector is array (1..10) of integer;
  type Matrix is array (1..10,1.. 5) of integer;

  task B is
    entry Data1(z: in Vector);
    entry Data2(z: out Matrix, e: out integer);
  end B;

  task body B is
    MT: Matrix;
    T: Vector;
    X: integer;
  begin

    accept Data1(z: in Vector) is
      T:= z;
    end Data1;

    accept Data2(z: out Matrix, e: out integer) is
      z:= MT;
      e:= x;
    end Data2;

  end B;

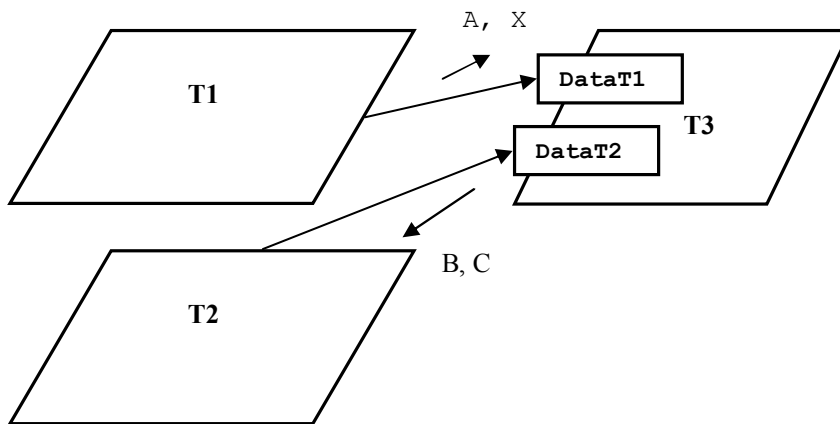
  task A;
  task body A is
    MA: Matrix;
    C: Vector;
    Y: integer;
  begin
    B.Data1(C);
    B.Data2(MA, y);
  end A;

begin
  null;
end Lab;
```

Завдання № 20

Розробити програму мовою Ада, в якій створюють та запускають на виконання три задачі T1, T2 і T3. За допомогою механізму рандеву передати зі задачі T1 в задачу T3 вектора A і X, а зі задачі T2 в задачу T3 передати два вектори B і C. Навести структурну схему взаємодії задач.

Відповідь :



```
procedure Lab is
  N: integer := 10;
  type Вектор is array(1..N) of integer;
  task T1;
  task T2;
  task T3 is
    entry DataT1(V1, V2: in Вектор);
    entry DataT2(V1, V2: in Вектор);
  end T3;
  task body T1 is
    A, X: Вектор;
  begin
    T3.DataT1(A, X);
  end T1;
  task body T2 is
    B, C: Вектор;
  begin
    T3.DataT2(B, C);
  end T2;
  task body T3 is
    VA, VB, VC, VX: Вектор;
  Begin
    accept DataT1(V1, V2: in Вектор) do
      VA := V1;
      VX := V2;
    end Data;
    accept DataT2(V1, V2: in Вектор) do
      VB := V1;
      VC := V2;
    end Data;
  end T3;
begin
  null;
end Lab;
```

Завдання № 21

Розробити програму мовою Ада, в якій створюють та запускають на виконання п'ять задач $T(1) \dots T(5)$. Використати задачний тип, за допомогою якого сформувати масив задач.

Відповідь :

```
-- задачний тип
task type ЗадачнТип;

task body ЗадачнТип is

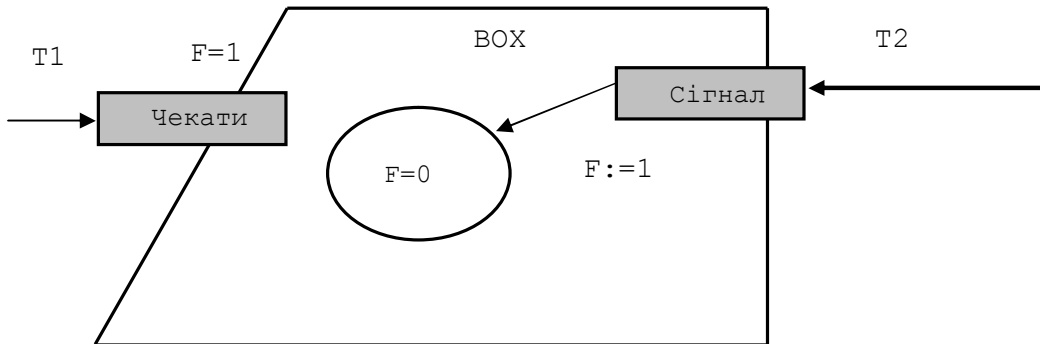
  Begin
    . . .
  end ЗадачнТип;

T: array (1..5) of ЗадачнТип;
```

Завдання № 22

Розробити захищений модуль, за допомогою якого забезпечити синхронізацію двох задач.

Відповідь :



```
-- захищений модуль
protected BOX is
  entry Чекати;
  procedure Сигнал;
private
  F : integer:= 0;      -- прапор
end Box;

-- тіло захищеного модуля
protected body BOX is

  entry Чекати when F =1 is
  begin
    null;
  end;

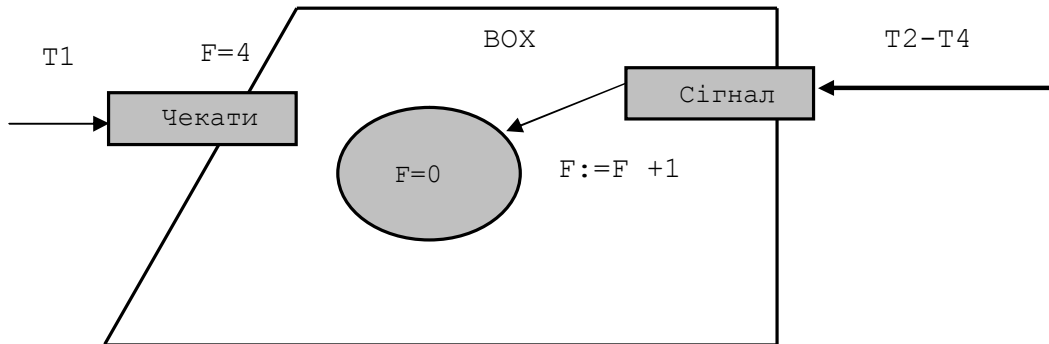
  procedure Signal is
  begin
    F := 1;
  end ;

end BOX;
```

Завдання № 23

Розробити захищений модуль, за допомогою якого забезпечити синхронізацію чотирьох задач. Одна задача чекає на сигнали від трьох задач.

Відповідь :



```
-- захищений модуль
protected BOX is
  entry Чекати;
  procedure Сигнал;
private
  F : integer:= 0;      -- прапор
end Box;

-- тіло захищеного модуля
protected body BOX is

  entry Чекати when F=4 is
  begin
    null;
  end;

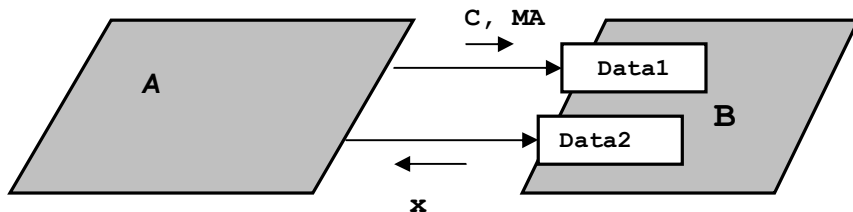
  procedure Signal is
  begin
    F := F + 1;
  end ;

end BOX;
```


Завдання № 24

Розробити програму мовою Ада, в якій створюються та запускаються на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В вектор С і матрицю МА, потім - прийняти скаляр x. Навести структурну схему взаємодії задач.

Відповідь :



```
procedure Lab is
```

```
    type Vector is array (1..10) of integer;
    type Matrix is array (1..10,1.. 5) of integer;
```

```
    task B is
```

```
        entry Data1(z: in Vector; M: in Matrix);
```

```
        entry Data2(e: out integer);
```

```
    end B;
```

```
    task body B is
```

```
        MT: Matrix;
```

```
        T: Vector;
```

```
        o: integer;
```

```
    begin
```

```
        accept Data1(z: in Vector; M: in Matrix) is
```

```
            T:= z;
```

```
            MT:= M;
```

```
        end Data1;
```

```
        accept Data2(e: out integer) is
```

```
            o:= e;
```

```
        end Data2;
```

```
    end B;
```

```
    task A;
```

```
    task body A is
```

```
        MA: Matrix;
```

```
        C: Vector;
```

```
        x: integer;
```

```
    begin
```

```
        B.Data1 (C,MA) ;
```

```
        B.Data2 (x) ;
```

```
    end A;
```

```
begin
```

```
    null;
```

```
end Lab;
```

Завдання № 25

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В матрицю МТ, вектор Е. Навести структурну схему взаємодії задач.

Відповідь :

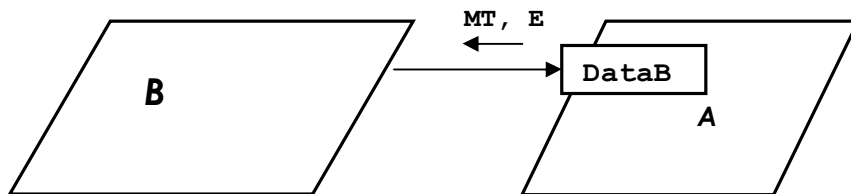
```
procedure Lab is
  type Вектор is array (1..5) of integer;
  type Matrix is array (1..5, 1 .. 5) of integer;

  task A is
    entry DataB(z: out Matrix; a: out Вектор);
  end A;
  task body A is
    MB: Matrix;
    q: Вектор;
  begin
    accept DataB (z: out Matrix; a: out Вектор) is
      z:= MB;
      a:= q;
    end Дані;

  end A;

  task B;
  task body B is
    MT : Matrix;
    E: Вектор;
  begin
    A.DataB(MT, E);
  end B;

begin
  null;
end Lab;
```



Завдання № 26

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі В в задачу А матрицю МТ, вектор Е. Навести структурну схему взаємодії задач.

Відповідь :

```
procedure Lab is

  type Вектор is array (1..5) of integer;
  type Matrix is array (1..5, 1 .. 5) of integer;

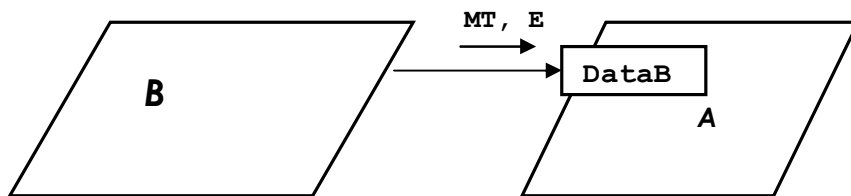
  task A is
    entry DataB(z: out Matrix; a: out Вектор);
  end A;
  task body A is
    MB: Matrix;
    q: Вектор;
  begin

    accept DataB (z: in Matrix; a: in Вектор) is
      MB:= z;
      q:= a;
    end Дані;

  end A;

  task B;
  task body B is
    MT : Matrix;
    E: Вектор;
  begin
    A.DataB(MT, E);
  end B;

begin
  null;
end Lab;
```



Завдання № 27

Розробити програму мовою Ада, в якій створюють та запускають на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В матрицю МТ та отримати від неї вектор Е. Навести структурну схему взаємодії задач.

Відповідь :

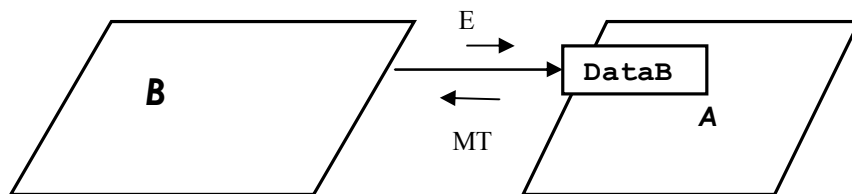
```
procedure Lab is
  type Вектор is array (1..5) of integer;
  type Matrix is array (1..5, 1 .. 5) of integer;

  task A is
    entry DataB(z: out Matrix; a: in Вектор);
  end A;
  task body A is
    MT: Matrix;
    q: Вектор;
  begin
    accept DataB (z: out Matrix; a: in Вектор) is
      z:= MT;
      q:= a;
    end Дані;

  end A;

  task B;
  task body B is
    MO : Matrix;
    E: Вектор;
  begin
    A.DataB(MO, E);
  end B;

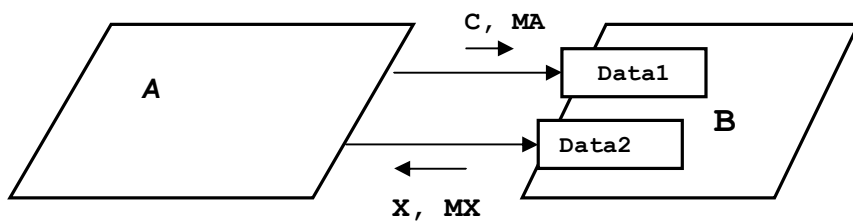
begin
  null;
end Lab;
```



Завдання № 28

Розробити програму мовою Ада, в якій створюються та запускаються на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В вектор С і матрицю МА, потім - прийняти скаляр x та матрицю МХ. Навести структурну схему взаємодії задач

Відповідь :



```
procedure Lab is
```

```
  type Vector is array (1..10) of integer;
  type Matrix is array (1..10,1.. 5) of integer;
```

```
  task B is
```

```
    entry Data1(z: in Vector; M: in Matrix);
    entry Data2(e: out integer; M: out Matrix);
```

```
  end B;
```

```
  task body B is
```

```
    MT: Matrix;
    T: Vector;
    o: integer;
```

```
  begin
```

```
    accept Data1(z: in Vector; M: in Matrix) is
      T:= z;
      MT:= M;
    end Data1;
```

```
    accept Data2(e: out integer; M: out Matrix) is
      e:= x;
      M:= MX;
    end Data2;
```

```
  end B;
```

```
  task A;
```

```
  task body A is
```

```
    MA: Matrix;
    C: Vector;
    x: integer;
```

```
  begin
```

```
    B.Data1(C,MA);
    B.Data2(x, MA);
```

```
  end A;
```

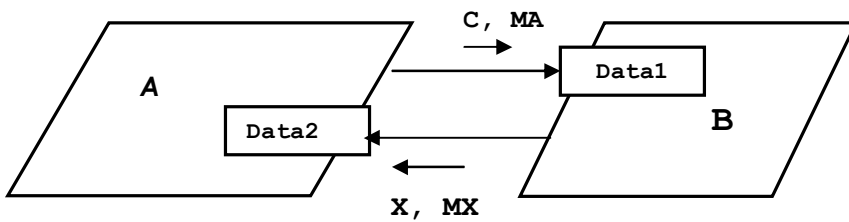
```
  begin
    null;
```

```
end Lab;
```

Завдання № 29

Розробити програму мовою Ада, в якій створюються та запускаються на виконання дві задачі А і В. За допомогою механізму рандеву передати зі задачі А в задачу В вектор С і матрицю МА, потім - прийняти скаляр x та матрицю МХ. Навести структурну схему взаємодії задач. Входи створити в різних задачах.

Відповідь :



procedure Lab is

```
type Vector is array (1..10) of integer;
type Matrix is array (1..10,1.. 5) of integer;
```

task B is

```
entry Data1(z: in Vector; M: in Matrix);
```

end B;

task body B is

```
MT: Matrix;
T: Vector;
o: integer;
```

begin

```
accept Data1(z: in Vector; M: in Matrix) is
```

```
T:= z;
```

```
MT:= M;
```

```
end Data1;
```

```
B.Data2(x, MX);
```

end B;

task A is

```
entry Data2(e: in integer; M: in Matrix);
```

end A;

task body A is

```
MA: Matrix;
```

```
C: Vector;
```

```
x: integer;
```

begin

```
B.Data1(C,MA);
```

```
accept Data2(e: in integer; M: in Matrix) is
```

```
x:= e;
```

```
MA:= M;
```

```
end Data2;
```

end A;

begin

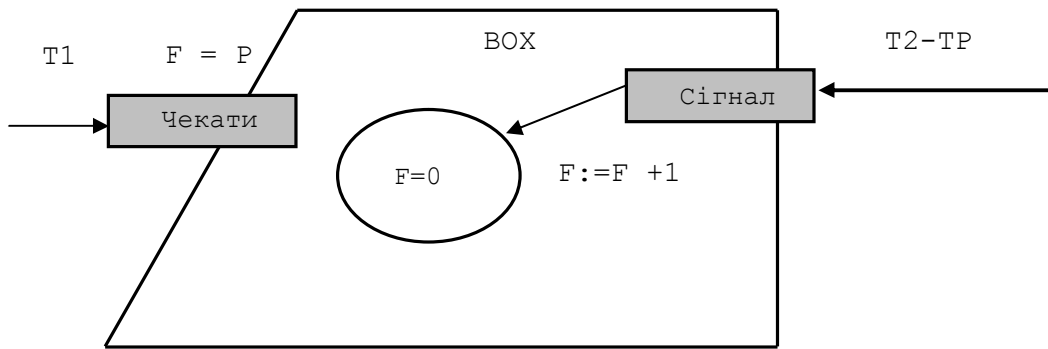
```
null;
```

end Lab;

Завдання № 30

Розробити захищений модуль, за допомогою якого забезпечити синхронізацію $(P + 1)$ задач. Одна задача чекає на сигнали від P задач.

Відповідь :



```
-- захищений модуль
protected BOX is
  entry Чекати;
  procedure Сигнал;
private
  F : integer := 0;      -- прапор
end BOX;

-- тіло захищеного модуля
protected body BOX is

  entry Чекати when F=P is
  begin
    null;
  end;

  procedure Signal is
  begin
    F := F + 1;
  end ;

end BOX;
```

Завдання № 31

Розробити програму мовою Java , в якій створюються та запускаються на виконання три потоки А, В і С. Встановити пріоритети потоків: 2 – для потоку А, 5 – для потоку В, 6 – для потоку С.

Відповідь :

```
class AT extends Thread {

    // перевизначення методу run()
    public void run() {
        . . .
    }
} // AT

class BT extends Thread {

    public void run() {
        ...
    }
} // BT

// головний потік
class Lab {

    // точка входу в основний клас
    public static void main(String args []) {

        // оголошення екземплярів потоків
        AT A = new AT();
        BT B = new BT();
        BT C = new BT();

        // пріоритети потоків
        A.Set_Priority(2);
        B.Set_Priority(5);
        C.Set_Priority(6);

        // запуск потоків
        A.start();
        B.start();
        C.start();

    } // main
} // Lab
```


Завдання № 32

Розробити програму мовою Java , в якій створюються та запускаються на виконання чотири потоки. Виконати формування та печать внутрішніх ідентифікаторів потоків від 1 до 4

Відповідь :

```
class AT extends Thread {
    int Tid;

    public AT(int x{
        Tid = x;          -- конструктор
    }

    // перевизначення методу run()
    public void run(){
        System.out.println(Tid);
    }
} // AT

// головний потік
class Lab {

    // точка входу в основний клас
    public static void main(String args []){

        // оголошення екземплярів потоків
        AT A = new AT(1);
        AT B = new AT(2);
        AT C = new AT(3);
        AT X = new AT(4);

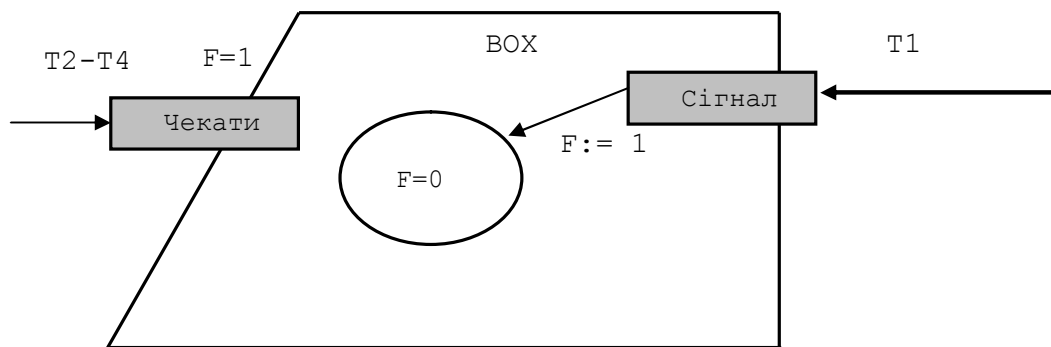
        // запуск потоків
        A.start();
        B.start();
        C.start();
        X.start();

    } // main
} // Lab
```

Завдання № 33

Розробити захищений модуль, за допомогою якого забезпечити синхронізацію трьох задач. Три задачі чекають на сигнал від однієї задачі.

Відповідь :



```
-- захищений модуль
protected BOX is
  entry Чекати;
  procedure Сигнал;  private
    F : integer:= 0;  -- прапор
end Box;

-- тіло захищеного модуля
protected body BOX is

  entry Чекати when F=1 is
  begin
    null;
  end;

  procedure Signal is
  begin
    F := 1;
  end;

end BOX;
```

Завдання № 34

Вирішити завдання взаємного виключення для двох задач за допомогою семафорів мови Ада

Відповідь :

```
procedure Lab is

    Buf: integer;           -- спільний ресурс
    S: Suspension_Object;   -- семафор

    task A;
    task body A is
    begin
        . . .
        Suspend_Until_True(S);
        Buf:= Buf - 1;       -- критична ділянка
        Set_True(S);
        . . .
    end A;

    task B;
    task body B is
    begin
        Suspend_Until_True(S);
        Buf:= Buf + 10;      -- критична ділянка
        Set_True(S);
        . . .
    end B;

begin
    Set_True(S);
end Lab;
```

Завдання № 35

Вирішити завдання синхронізації двох задач за допомогою семафорів мови Ада

Відповідь :

```
procedure Lab is

    S: Suspension_Object;           -- семафор

    task A;
    task body A is
    begin
        . . .
        Suspend_Until_True(S);      -- очікування на подію
        . . .
    end A;

    task B;
    task body B is
    begin
        . . .
        Set_True(S);                -- сигнал про подію
        . . .
    end B;

begin
    null;
end Lab;
```

Завдання № 36

Вирішити завдання синхронізації двох задач за допомогою семафорів Win32

Відповідь :

```
procedure Lab is

    S: HANDLE;          -- семафор

    task A;
    task body A is
    begin
        . . .
        WaitForSingleObject(S,infinite);    -- очікування на подію
        . . .
    end A;

    task B;
    task body B is
        . . .
        ReleaseSemaphore(S,1);              -- сигнал про подію
        . . .
    end B;

begin

    S:=CreateSemaphore(Null,1,0,NULL);

end Lab;
```

Вирішити завдання взаємного виключення для двох задач за допомогою критичних секцій Win32

Відповідь :

```
procedure Lab is

    BUF: integer:=0;           -- спільний ресурс
    S: CRITICAL_SECTION;      -- семафор

    task A;
    task body A is
    begin
        . . .
        EnterCriticalSection(S);
        BUF:= BUF + 1;         -- критична ділянка
        LeaveCriticalSection(S);
        . . .
    end A;

    task B;
    task body B is
    begin
        . . .
        EnterCriticalSection(S);
        BUF:= BUF - 100;       -- критична ділянка
        LeaveCriticalSection(S);
        . . .
    end B;

begin
    null;

end Lab;
```

Завдання № 38

Вирішити завдання взаємного виключення для двох задач за допомогою мютексів Win32

Відповідь :

```
procedure Lab is

    BUF: integer:=0;                -- спільний ресурс
    M: Handle;                      -- мютекс

    task A;
    task body A is
    begin
        . . .
        WaitForSingleObject(M, infinite);
        BUF:= BUF + 200;            -- критична ділянка
        ReleaseMutex(M) ;
        . . .
    end A;

    task B;
    task body B is
    begin
        . . .
        WaitForSingleObject(M, infinite);
        BUF:= BUF - 100;           -- критична ділянка
        ReleaseMutex(M) ;
        . . .
    end B;

begin
    M:= CreateMutex(...);

end Lab;
```

Вирішити завдання синхронізації для трьох задач за допомогою семафорів Win32. Три задачі A , B і E чекають на сигнал від однієї задачі C

Відповідь :

```
procedure Lab is

    S: HANDLE;          -- семафор

    task A;
    task body A is
    begin
        . . .
        WaitForSingleObject(S,infinite);    -- очікування на подію
        . . .
    end A;

    task B ;
    task body B is
    begin
        . . .
        WaitForSingleObject(S,infinite);    -- очікування на подію
        . . .
    end B;

    task E;
    task body E is
    begin
        . . .
        WaitForSingleObject(S,infinite);    -- очікування на подію
        . . .
    end E;

    task body C is
        . . .
        ReleaseSemaphore(S,3);              -- сигнал про подію
        . . .
    end C;

begin

    S:=CreateSemaphore(Null,3,0,NULL);

end Lab;
```


Завдання № 40

Вирішити завдання синхронізації для трьох задач за допомогою семафорів Win32. Дві задачі А і В чекають на сигнал від однієї задачі С

Відповідь :

```
procedure Lab is

    S: HANDLE;          -- семафор

    task A;
    task body A is
    begin
        . . .
        WaitForSingleObject(S,infinite);    -- очікування на подію
        . . .
    end A;

    task B;
    task body B is
    begin
        . . .
        WaitForSingleObject(S,infinite);    -- очікування на подію
        . . .
    end B;

    task body C is
    begin
        . . .
        ReleaseSemaphore(S,2);              -- сигнал про подію
        . . .
    end C;

begin

    S:=CreateSemaphore(Null,0,2,NULL);

end Lab;
```