

Cours Algorithmique des systèmes parallèles et distribués
 Exercices
 Série :PlusCal pour la programmation répartie ou concurrente (I)
 par Dominique Méry
 11 février 2026

Exercice 1 (*pluscaltut1.tla*)

Etudier, compléter et analyser le programme PlusCal suivant :

----- MODULE *pluscaltut1q* -----

EXTENDS Integers , Sequences , TLC, FiniteSets

(*

--wf

--algorithm Tut1 {

variables x = 0;

process (one = 1)

{

 A: assert x \in ???;

 x := x - 1;

 B: assert x \in ???? ;

 x := x * 3;

 BB: assert x \in ???;

};

process (two = 2)

{

 C: assert x \in ???;

 x := x + 1;

 D:

 assert x \in ??;

};

}

end algorithm;

*)

safepc == pc[1] = "Done" /\ pc[2] = "Done" => ??

=====

Exercice 2 (*pluscaltut2.tla*)

Etudier, compléter et analyser le programme PlusCal suivant :

```

----- MODULE pluscaltut2q -----
EXTENDS Integers , Sequences , TLC, FiniteSets

(*
--algorithm Tut2 {
variables x = 0;

process (one = 1)

variables temp
{

A:
temp := x + 1;

x := temp;

};

process (two = 2)

variables temp
{
B:
temp := x + 1;

x:= temp;

};

}
end algorithm;

*)

saferpc == pc[1] = "Done" /\ pc[2] = "Done" => ??
=====
```

Exercice 3 (*pluscaltut3.tla*)

Etudier le programme PlusCal suivant :

```

----- MODULE pluscaltut3q -----
EXTENDS Integers , Sequences , TLC, FiniteSets
(*
--algorithm Tut3 {
```

```

variables x = 0;

process (one = 1)
{
    A:
        x := x + 1;
    B:
        await x = 1;
    C:
        print <<"x=",x>>;
};

process (two = 2)
{
    D:
        await x = 1;
    E:
        assert x = 1;
    F:
        x := x -2;
};

}

end algorithm;
*)

=====

```

Exercice 4 pluscaltut4.tla

Ecrire un programme PlusCal qui traduit le protocole suivant :

- S envoie une valeur val à R
- R reçoit la même valeur val

Exercice 5 pluscaltut5.tla

Ecrire un programme PlusCal qui calcule la fonction factorielle de la façon suivante :

- Un processus P1 calcule $1 \times 2 \times 3 \dots \times k_1$
- Un processus P2 calcule $k_2 \times (k_2+1) \times \dots \times N$
- Les processus stoppent quand la condition $k_1 < k_2$ est fausse

Exercice 6 pluscaltut6.tla

Ecrire un programme PlusCal qui calcule la fonction L^K la façon suivante :

- Un processus P_1 calcule $L \times \dots \times L$ k_1 fois.
- Un processus P_2 calcule $L \times \dots \times L$ k_2 fois.
- Les processus P_3 stoppent quand la condition $k_1 + K_2 < L$ est fausse