

Compte Rendu TP3

Question1

```
taches(Taches):-
    Taches = [](tache(3, [],m1, _),
    tache(8, [],m1, _),
    tache(8, [4,5], m1, _),
    tache(6, [],m2, _),
    tache(3, [1],m2, _),
    tache(4, [1,7], m1, _),
    tache(8, [3,5], m1, _),
    tache(6, [4],m2, _),
    tache(6, [6,7], m2, _),
    tache(6, [9,12], m2, _),
    tache(3, [1],m2, _),
    tache(6, [7,8], m2, _)).
```

Test

```
[eclipse 2]: taches(T).

T = [](tache(3, [], m1, _181), tache(8, [], m1, _186), tache(8, [4, 5], m1, _191), tache(6, [], m2, _200), tache(3, [1], m2, _205), tache(4, [1, 7], m1, _210), tache(8, [3, 5], m1, _215), tache(6, [4], m2, _220), tache(6, [6, 7], m2, _225), tache(6, [9, 12], m2, _230), tache(3, [1], m2, _235), tache(6, [7, 8], m2, _240)).
Yes (0.00s cpu)
```

Question2

```
ecrit_taches(Taches):-
    dim(Taches,Dim),
    D is Dim[1],
    (for(I,1,D),
    param(Taches)
    do
        Var is Taches[I],
        writeln(Var)
    ).
```

Test

```
taches(Taches),ecrit_taches(Taches).
[eclipse 3]: taches(Taches),ecrit_taches(Taches).
tache(3, [], m1, _243)
tache(8, [], m1, _248)
tache(8, [4, 5], m1, _253)
tache(6, [], m2, _262)
tache(3, [1], m2, _267)
tache(4, [1, 7], m1, _274)
tache(8, [3, 5], m1, _283)
tache(6, [4], m2, _292)
tache(6, [6, 7], m2, _299)
tache(6, [9, 12], m2, _308)
tache(3, [1], m2, _317)
tache(6, [7, 8], m2, _324)

Taches = [](tache(3, [], m1, _243), tache(8, [], m1, _248), tache(8, [4, 5], m1, _253), tache(6, [], m2, _262), tache(3, [1], m2, _267), tache(4, [1, 7], m1, _274), tache(8, [3, 5], m1, _283), tache(6, [4], m2, _292), tache(6, [6, 7], m2, _299), tache(6, [9, 12], m2, _308), tache(3, [1], m2, _317), tache(6, [7, 8], m2, _324)).
Yes (0.00s cpu)
```

Question3

```
domaines(Taches,Fin):-
    dim(Taches,Dim),
    D is Dim[1],
    (for(I,1,D),
    param(Taches,Fin)
    do
        tache(_, _ ,M, D) is Taches[I],
        M>=0,
        D#>=0,
        D#<Fin
    ).
```

Test

```
[eclipse 4]: taches(Taches),domaines(Taches,5).
```

```
Taches = [(tache(3, [], m1, _439{0 .. 5}), tache(8, [], m1, _576{0 .. 5}), tache(8, [4, 5], m1, _713{0 .. 5}), tache(6
Yes (0.00s cpu)
```

Question4

```
getVarList(Taches,Fin,[Fin|Liste]):-
    dim(Taches,Dim),
    D is Dim[1],
    (for(I,1,D),
    param(Taches),
    fromto([],In,Out,Liste)
    do
        tache(_, _, Machine, Debut) is Taches[I],
        Out = [Debut|In]
    ).
```

Test

```
[eclipse 5]: taches(Taches),domaines(Taches,5),getVarList(Taches,5,List).
```

```
Taches = [(tache(3, [], m1, _513{0 .. 5}), tache(8, [], m1, _650{0 .. 5}), tache(8, [4, 5], m1, _787{0 .. 5}), tache(6
List = [5, _2020{0 .. 5}, _1883{0 .. 5}, _1746{0 .. 5}, _1609{0 .. 5}, _1472{0 .. 5}, _1335{0 .. 5}, _1198{0 .. 5}, _10
Yes (0.00s cpu)
```

Question5

```
solve(Taches,Fin):-
    taches(Taches),
    FinTotal = 0,
    domaines(Taches,Fin),
    precedences(Taches),
    conflicts(Taches),
    getVarList(Taches,Fin,Liste),
    labeling(Liste),
    ecrit_taches(Taches).
```

Test

```
[eclipse 6]: taches(Taches),solve(Taches,Fin).
lists.eco loaded in 0.00 seconds
tache(3, [], m1, 0)
```

```
tache(8, [], m1, 0)
tache(8, [4, 5], m1, 6)
tache(6, [], m2, 0)
tache(3, [1], m2, 3)
tache(4, [1, 7], m1, 22)
tache(8, [3, 5], m1, 14)
tache(6, [4], m2, 6)
tache(6, [6, 7], m2, 26)
tache(6, [9, 12], m2, 32)
tache(3, [1], m2, 3)
tache(6, [7, 8], m2, 22)

Taches = [](tache(3, [], m1, 0), tache(8, [], m1, 0), tache(8, [4, 5], m1, 6), tache(6, [], m2, 0), tache(3, [1], m2, 3)
Fin = 32
```

Question6

```
precedences(Taches):-
    dim(Taches,Dim),
    D is Dim[1],
    (for(I,1,D),
    param(Taches)
    do
        tache(_, Noms,_, Debut) is Taches[I],
        (foreach(Nom,Noms),
        param(Taches,Debut)
        do
            tache(DureePrec, _, _, DebutPrec) is Taches[Nom],
            Debut#>= DureePrec+DebutPrec
        )
    ).
```

Test

```
[eclipse 7]: taches(Taches),solve(Taches,Fin).
tache(3, [], m1, 0)
tache(8, [], m1, 0)
tache(8, [4, 5], m1, 6)
tache(6, [], m2, 0)
tache(3, [1], m2, 3)
tache(4, [1, 7], m1, 22)
tache(8, [3, 5], m1, 14)
tache(6, [4], m2, 6)
tache(6, [6, 7], m2, 26)
tache(6, [9, 12], m2, 32)
tache(3, [1], m2, 3)
tache(6, [7, 8], m2, 22)

Taches = [](tache(3, [], m1, 0), tache(8, [], m1, 0), tache(8, [4, 5], m1, 6), tache(6, [], m2, 0), tache(3, [1], m2, 3)
Fin = 32
```

Question7

```
conflicts(Taches):-
    dim(Taches,Dim),
    D is Dim[1],
    (for(I,1,D),
    param(Taches,D)
    do
        tache(Duree, _, Machine, Debut) is Taches[I],
        I2 is I+1,
        (for(J,I2,D),
        param(Taches,Debut,Duree)
        do
            tache(Duree2, _, Machine2, Debut2) is Taches[J],
```

```
(Machine &\= Machine2) => ((Debut#>=Debut2+Duree2) or (Debut+Duree#=>Debut2))
)
).
```

Test

```
[eclipse 10]: taches(T),conflicts(T).
```

```
T = [](tache(3, [], m1, _1336{-1.0Inf .. 1.0Inf}), tache(8, [], m1, _1319{-1.0Inf .. 1.0Inf}), tache(8, [4, 5], m1, _25
```

Test

```
[eclipse 20]: solve(Taches,Fin).
tache(3, [], m1, 0)
tache(8, [], m1, 0)
tache(8, [4, 5], m1, 6)
tache(6, [], m2, 0)
tache(3, [1], m2, 3)
tache(4, [1, 7], m1, 22)
tache(8, [3, 5], m1, 14)
tache(6, [4], m2, 6)
tache(6, [6, 7], m2, 26)
tache(6, [9, 12], m2, 32)
tache(3, [1], m2, 3)
tache(6, [7, 8], m2, 22)

Taches = [](tache(3, [], m1, 0), tache(8, [], m1, 0), tache(8, [4, 5], m1, 6), tache(6, [], m2, 0), tache(3, [1], m2, 3
Fin = 38
Yes (0.01s cpu, solution 1, maybe more) ?
```

Question 8

```
[eclipse 36]: solve(Taches,Fin).
tache(3, [], m1, 0)
tache(8, [], m1, 29)
tache(8, [4, 5], m1, 9)
tache(6, [], m2, 0)
tache(3, [1], m2, 6)
tache(4, [1, 7], m1, 25)
tache(8, [3, 5], m1, 17)
tache(6, [4], m2, 12)
tache(6, [6, 7], m2, 31)
tache(6, [9, 12], m2, 37)
tache(3, [1], m2, 9)
tache(6, [7, 8], m2, 25)

Taches = [](tache(3, [], m1, 0), tache(8, [], m1, 29), tache(8, [4, 5], m1, 9), tache(6, [], m2, 0), tache(3, [1], m2,
Fin = 43
Yes (0.02s cpu, solution 1, maybe more) ?
[eclipse 37]: Fin#<43,solve(Taches,Fin).
```

```
No (0.01s cpu)
```

La première solution offerte se finit en 43 unités de temps, par conséquent nous cherchons une solution qui se finit en moins de temps que celle-ci, puisque nous n'en trouvons pas, nous savons que nous avons la plus rapide