

Compte Rendu TD7

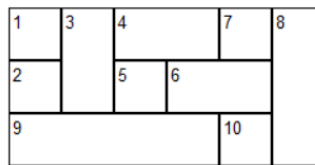
Andrinirina Gatien

1 ère Etape

```
:- use_module(gmgraph).
```

```
dessin_boites :- gr_init,  
gr_rect_pos(1, 100, 100, 40, 40),  
gr_rect_pos(2, 100, 140, 40, 40),  
gr_rect_pos(3, 140, 100, 40, 80),  
gr_rect_pos(4, 180, 100, 80, 40),  
gr_rect_pos(7, 260, 100, 40, 40),  
gr_rect_pos(8, 300, 100, 40, 120),  
gr_rect_pos(5, 180, 140, 40, 40),  
gr_rect_pos(6, 220, 140, 80, 40),  
gr_rect_pos(9, 100, 180, 160, 40),  
gr_rect_pos(10, 260, 180, 40, 40).
```

```
?- dessin_boites.  
true.  
?-
```



2 ème Etape

```
voisins(1, [2,3]).  
voisins(2, [1,3,9]).  
voisins(3, [1,2,4,5,9]).  
voisins(4, [3,5,6,7]).  
voisins(5, [3,4,6,9]).  
voisins(6, [4,5,7,8,9,10]).  
voisins(7, [4,6,8]).  
voisins(8, [6,7,10]).  
voisins(9, [2,3,5,6,10]).  
voisins(10, [6,8,9]).
```

```
?- voisins(4,L).  
L = [3, 5, 6, 7].  
?-
```

3 ème Etape

```
couleurs(yellow).  
couleurs(red).  
couleurs(green).  
couleurs(blue).  
couleurs(black).
```

4 ème Etape

```
dessin_couleurs([],_).
```

```

dessin_couleurs([Couleur | Q], N) :-
    gr_rect_couleur(N,Couleur),
    N1 is N+1,
    dessin_couleurs(Q,N1).

?- dessin_boites,dessin_couleurs([red,yellow,blue,green,red,yellow,green,orange,black,gray],1).
true.
?-

```



5 ème Etape

```

solutions(ListeC) :-
    couleurs(C1 ),
    couleurs(C2 ),
    couleurs(C3 ),
    couleurs(C4 ),
    couleurs(C5 ),
    couleurs(C6 ),
    couleurs(C7 ),
    couleurs(C8 ),
    couleurs(C9 ),
    couleurs(C10),
    ListeC = [C1, C2, C3, C4, C5, C6, C7, C8, C9, C10]
.

```

6 ème Etape

```

trouveCouleurs(_, [], []).
trouveCouleurs(LC, [ V | QV], [ C | R ]) :-
    nth1(V, LC, C),
    trouveCouleurs(LC,QV,R)
.
?- trouveCouleurs([red, blue, orange, red,dark], [1,2,3], R).
R = [red, blue, orange] ;
false.
?- 

```

7 ème Etape

```

voisins_ok(N, LC) :-
    voisins(N,LV),
    trouveCouleurs(LC,LV,LX),
    nth1(N,LC,X),
    \+member(X,LX)
.
?- voisins_ok(7,[yellow, red, green, yellow, red, green, red, yellow,yellow,red]).
true.
?- voisins_ok(4,[yellow, red, yellow, yellow, red, green, red, yellow,yellow,red]).
false.
?-

```

8 ème Etape

```

solution(ListeC) :-
    couleurs(C1 ), % on génère la couleur C1 du pays 1

```

```

couleurs(C2 ), % la couleur C2 du pays 2
couleurs(C3 ),
couleurs(C4 ), % cf générateur logique de cryptarithme
couleurs(C5 ), % cours & TD
couleurs(C6 ),
couleurs(C7 ),
couleurs(C8 ),
couleurs(C9 ),
couleurs(C10),
ListeC = [C1, C2, C3, C4, C5, C6, C7, C8, C9, C10],
% on fabrique la liste solution des couleurs
voisins_ok(1 , ListeC),
voisins_ok(2 , ListeC),
voisins_ok(3 , ListeC),
voisins_ok(4 , ListeC),
voisins_ok(5 , ListeC),
voisins_ok(6 , ListeC),
voisins_ok(7 , ListeC),
voisins_ok(8 , ListeC),
voisins_ok(9 , ListeC),
voisins_ok(10 , ListeC).

```

```

?- dessin_boites(solution(L),dessin_couleurs(L,1)).
L = [yellow, red, green, yellow, red, green, red, yellow, yellow|...] ;
L = [yellow, green, red, yellow, green, red, green, yellow, yellow|...] ;
L = [red, yellow, green, red, yellow, green, yellow, red, red|...] ;
L = [red, green, yellow, red, green, yellow, green, red, red|...] ;
L = [green, yellow, red, green, yellow, red, yellow, green, green|...] ;
L = [green, red, yellow, green, red, yellow, red, green, green|...] ;
false.

```

?-



Pour trouver le nombre de solutions, on met dans la console :

```
?- findall(LC,solution(LC),L),length(L,R),write(R).
```

Pour 2 couleurs

```
couleurs(yellow).
couleurs(red).
```

```

?-
% c:/users/kenzo/onedrive/documents/prolog/td7/td7 compiled 0.00 sec, -1 clauses
?- findall(LC,solution(LC),L),length(L,R),write(R).
0
L = [].
R = 0.
?-

```

0 solution

Pour 3 couleurs

```
couleurs(yellow).
couleurs(red).
couleurs(green).
```

```

?-
% c:/users/kenzo/onedrive/documents/prolog/td7/td7 compiled 0.00 sec, 1 clauses
?- findall(IC,solution(IC),L),length(L,R),write(R).
6
L = [[yellow, red, green, yellow, red, green, red, yellow|...], [yellow, green, red, yellow, green, red, green|...], [red, yellow, green, red, yellow, green|...], [red, green, yellow, red, green|...], [green, yellow, red, green|...], [green, red, yellow|...]],
R = 6.

?-

```

6 solutions

Pour 4 couleurs

```

couleurs(yellow).
couleurs(red).
couleurs(green).
couleurs(blue).

```

```

?-
% c:/users/kenzo/onedrive/documents/prolog/td7/td7 compiled 0.00 sec, 1 clauses
?- findall(IC,solution(IC),L),length(L,R),write(R).
1632
L = [[yellow, red, green, yellow, red, green, red, yellow|...], [yellow, red, green, yellow, red, green, red|...], [yellow, red, green, yellow, red, green|...], [yellow, red, green, yellow, red|...], [yellow, red, green, yellow|...], [yellow, red, green|...], [yellow, red|...], [yellow|...], [...|...]|...],
R = 1632.

?-

```

1632 solutions

Pour 5 couleurs

```

couleurs(yellow).
couleurs(red).
couleurs(green).
couleurs(blue).
couleurs(black).

```

```

% c:/users/kenzo/onedrive/documents/prolog/td7/td7 compiled 0.00 sec, 1 clauses
?- findall(IC,solution(IC),L),length(L,R),write(R).
77220
L = [[yellow, red, green, yellow, red, green, red, yellow|...], [yellow, red, green, yellow, red, green, red|...], [yellow, red, green, yellow, red, green|...], [yellow, red, green, yellow, red|...], [yellow, red, green, yellow|...], [yellow, red, green|...], [yellow, red|...], [yellow|...], [...|...]|...],
R = 77220.

?-

```

77220 solutions