

T.P. Prolog n°4

Exercice 1 : Mots croisés sans case noire

a) Ecrire un prédicat qui affiche les solutions d'une grille de mots croisés 3x3 contenant 3 mots horizontaux et 3 mots verticaux, sans aucune case noire. On fera en sorte que les mots de la grille soient tous différents.

Exemple :

| | | |
|---|---|---|
| a | d | o |
| g | i | n |
| e | s | t |

Pour résoudre le problème on utilisera le fichier `mots3lettres.pl` contenant tous les mots de trois lettres de la langue française. **Ne pas modifier ce fichier.** Pour faire en sorte qu'il soit chargé par Prolog au moment du chargement de votre programme, placez en tête de votre programme la ligne suivante :

```
:- consult('mots3lettres.pl').
```

b) Améliorez votre solution de sorte que les deux diagonales correspondent aussi à un mot, c'est le cas de l'exemple ci-dessus.

Exercice 2 : Labyrinthe

Soit le labyrinthe ci-contre.

Ecrire le prédicat `solution/1` qui détermine un chemin pour aller de la case 1 à la case 25. Les passages entre les cases seront représentés par des faits de la forme :

```
passage(1,2).
```

```
passage(1,6).
```

```
passage(2,7).
```

...

| | | | | |
|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 |
| 6 | 7 | 8 | 9 | 10 |
| 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 |

Récupérez fichier `labyrinthe.pl` qui contient déjà ces faits.

Lorsque l'on est dans la case `i`, l'idée est de passer dans un état `j` par le biais d'un fait `passage(i,j)`. Une fois passé dans la case `j`, le problème redevient le même : il faut aller de `j` à 25 ! Il faut bien sûr mémoriser le chemin et ne pas repasser par une case déjà visitée. La récursivité s'arrête lorsque l'on est dans l'état 25. A ce moment, la solution est exactement le chemin déjà parcouru. Pour cela, on utilise un prédicat intermédiaire `solution/4` qui va mémoriser le chemin déjà parcouru (accumulateur). Commencez par écrire ce prédicat :

```
solution(CaseCourante,But,CheminDejaParcoursu,Solution) que l'on appellera avec cette requête :
```

```
?- solution(1,25,[1],Solution).
```

Question subsidiaire : dessiner un labyrinthe vide en affichant ligne par ligne le labyrinthe. Chaque case est représentée par le caractère '_' s'il y a un mur vers le bas ou par un espace sinon. Entre chaque case, afficher le caractère '|' s'il y a un mur ou un espace sinon.

Exemple :

```
  _ _ | _ _ _  
| | | | _ |  
| | | | _ |  
| _ | | | _ |  
| _ _ | | _ _
```