Université Pierre-Mendès-France
Licence MIASS, 2° année - L2 INF f4

T.P. Prolog n°3

1) Prenons le train

On dispose de faits représentant des trajets de trains entre deux villes. Par exemple, le fait suivant indique qu'il existe un train allant de Grenoble à Paris partant à 8h et arrivant à 11h :

train(grenoble,paris,8,11).

Récupérez le fichier **horaires_trains.pl** contenant une liste de ces faits.

a) Ecrire le prédicat **liaison**/3 qui détermine les chemins possibles pour aller d'une ville à une autre, sans tenir compte des horaires des correspondances. Par exemple :

?- liaison(paris,gieres,S).
S=[[paris,grenoble],[grenoble,gieres]] ;
S=[[paris,chambery],[chambery,grenoble],[grenoble,gieres]]
...

Attention, il faut éviter de boucler indéfiniment en repassant plusieurs fois par la même ville. Vous éviterez aussi les aller-retours inutiles comme :

[[paris,grenoble][grenoble,chambery],[chambery,grenoble],[grenoble,gieres]]

b) Même question, en tenant compte cette fois des horaires des correspondances, pour un trajet dans la même journée. Le prédicat devra retourner villes et horaires associés :

?-voyage(paris,gieres,S).
S=[[paris,grenoble,7,10],[grenoble,gieres10,11]] ;
S=[[paris,chambery,9,12],[chambery,grenoble,12,13],[grenoble,gieres,14,15]]

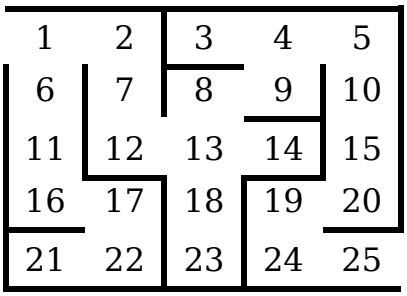
c) Ecrire le prédicat plusrapide/4 qui calcule le voyage le plus rapide entre deux villes.

?- plusrapide(grenoble,paris,Voyage,Duree).

Voyage = [[grenoble, lyon, 7, 8], [lyon, paris, 8, 9]],
Duree = 2

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).
...
```

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 qui va mémoriser le chemin déjà parcouru. Commencer donc par écrire le prédicat :

solution(CaseCourante,But,CheminDejaParcouru,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 un mur ou un espace sinon.

Exemple :

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