

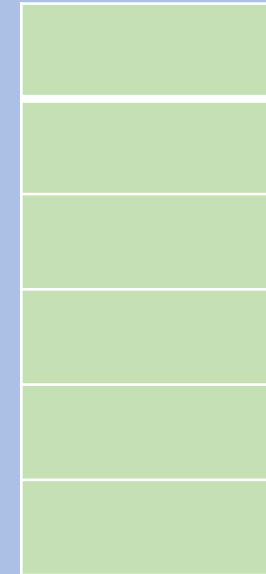
# Méthode pour trouver la solution du labyrinthe suivant

1	2	1	1	1	1
1	0	0	0	0	1
1	0	1	0	1	1
1	0	1	0	0	1
1	0	0	1	0	1
1	1	1	1	3	1

# Etape 1 :

On va utiliser une copie de notre labyrinthe ( afin de ne pas modifier le labyrinthe d'origine) et créer une pile vide.

1	2	1	1	1	1
1	0	0	0	0	1
1	0	1	0	1	1
1	0	1	0	0	1
1	0	0	1	0	1
1	1	1	1	3	1



Une case sera dite libre si elle n'a pas encore été visité et si ce n'est pas un mur.

# Etape 2 :

1. On examine la case de départ et on la marque comme visitée (4)
2. On regarde la liste de ses cases voisines libres : [(1, 1)].  
Comme cette liste n'est pas vide, on empile la case examinée et on examine sa première voisine libre, ici (1, 1)

1	4	1	1	1	1
1	0	0	0	0	1
1	0	1	0	1	1
1	0	1	0	0	1
1	0	0	1	0	1
1	1	1	1	3	1

(0, 1)

# Etape 3 :

1. On examine la case (1, 1) et on la marque comme visitée (4)
2. On regarde la liste de ses cases voisines libres: [(2, 1), (1, 2)].  
Comme cette liste n'est pas vide, on empile la case examinée (1, 1)  
et on examine sa première voisine libre ici : (2, 1)

1	4	1	1	1	1
1	4	0	0	0	1
1	0	1	0	1	1
1	0	1	0	0	1
1	0	0	1	0	1
1	1	1	1	3	1

(1, 1)
(0, 1)

# Etape 4 :

1. On examine la case (2, 1) et on la marque comme visitée (4)
2. On regarde la liste de ses cases voisines libres: [(3, 1)].  
Comme cette liste n'est pas vide, on empile la case examinée (2, 1)  
et on examine sa première voisine libre (3, 1)

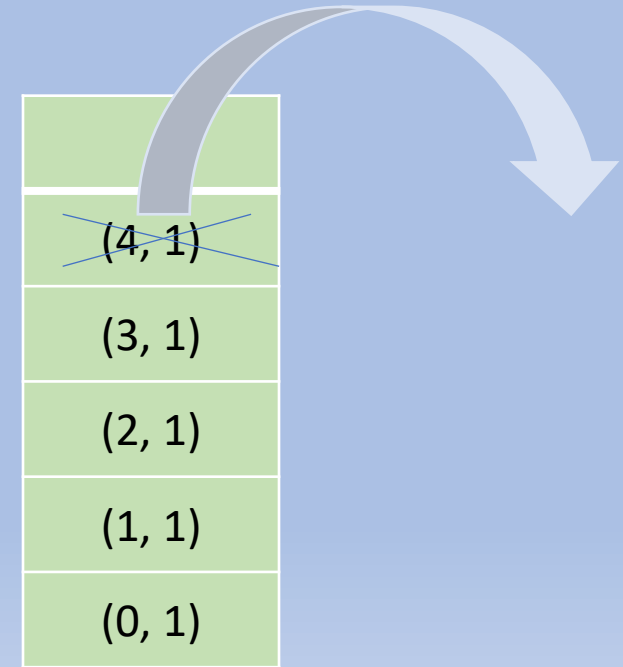
1	4	1	1	1	1
1	4	0	0	0	1
1	4	1	0	1	1
1	0	1	0	0	1
1	0	0	1	0	1
1	1	1	1	3	1

(2, 1)
(1, 1)
(0, 1)

# Etape 5 :

1. Et ainsi de suite jusqu'à ce que l'on tombe sur une impasse ...
2. On examine la case (4, 2) et on la marque comme visitée (4)
3. On regarde la liste de ses cases voisines libres: [],  
Comme cette liste est vide, on dépile la pile.

1	4	1	1	1	1
1	4	0	0	0	1
1	4	1	0	1	1
1	4	1	0	0	1
1	4	4	1	0	1
1	1	1	1	3	1



# Etape 6 :

1. On examine les prochaines voisines libres de la case dépilée ici (4, 1)
2. Et on examine , ses case voisines libres, comme il n'y en a pas, on dépile la suivante

1	4	1	1	1	1
1	4	0	0	0	1
1	4	1	0	1	1
1	4	1	0	0	1
1	4	4	1	0	1
1	1	1	1	3	1

(3, 1)
(2, 1)
(1, 1)
(0, 1)

# Etape 7 :

Et ainsi de suite, jusqu'à se retrouver dans la situation on a une case ayant des cases voisines non visitée ici la case (1, 1).

1	4	1	1	1	1
1	4	0	0	0	1
1	4	1	0	1	1
1	4	1	0	0	1
1	4	4	1	0	1
1	1	1	1	3	1

(1, 1)
(0, 1)



# Etape 8 :

1. On examine la case (1, 1)
2. On regarde la liste de ses cases voisines libres: [(1, 2)],  
Comme cette liste n'est pas vide, on examine sa voisine (1, 2)  
et on regarde les listes des cases voisines libres.

1	4	1	1	1	1
1	4	0	0	0	1
1	4	1	0	1	1
1	4	1	0	0	1
1	4	4	1	0	1
1	1	1	1	3	1

(1, 1)
(0, 1)

# Etape 9 :

1. Et ainsi de suite, jusqu'à temps que l'on arrive sur la case de sortie
2. Et dans ce cas, le chemin « apparaît » dans la pile (de bas en haut)

1	4	1	1	1	1
1	4	4	4	0	1
1	4	1	4	1	1
1	4	1	4	4	1
1	4	4	1	4	1
1	1	1	1	3	1

(5, 4)
(4, 4)
(3, 4)
(3, 3)
(2, 3)
(1, 3)
(1, 2)
(1, 1)
(0, 1)