

Ariane, quand tu nous tiens

3 février 2018

1 le background

Ariane, fille du roi Minos, s'est laissée séduire par Thésée et lui fournit un fil magique qui lui permettra de sortir du labyrinthe de Dédale, dans lequel est enfermé le fameux Minotaure (qui est par ailleurs le demi-frère d'Ariane). Comme tout bon héros mâle grec, Thésée tue le monstre, sort du labyrinthe et remercie Ariane en l'abandonnant sur l'île de Dia, quoi que certaines versions présentent cet abandon comme non volontaire.

L'histoire ne dit pas comment Thésée s'est servi du fil mais on peut néanmoins conjecturer que malgré son machisme évident, il avait une once de cervelle. Il avait donc certainement constaté que s'il recroisait le fil, ce n'était pas top et qu'il devait revenir en arrière jusqu'au lieu du croisement. Mais cela ne suffit pas comme stratégie pour sortir d'un labyrinthe et comme tous les héros soit disant malins, une voix divine lui a certainement soufflé la solution.

2 l'énoncé

Malheureusement pour vous, les Dieux grecs sont depuis longtemps morts et enterrés et aucune aide mystique ne va venir vous conseiller. Votre objectif est de retrouver l'astuce de Thésée pour tracer votre chemin dans un labyrinthe. Ce labyrinthe est représenté par un rectangle à cases, avec des murs épais. La figure 1 en donne un exemple. Le labyrinthe ne comporte qu'une seule sortie.

```
#####  
# # # # # #  
# # # # # #  
### ### # ###  
# # # # # #  
#####
```

FIGURE 1 – Exemple de labyrinthe avec une sortie

Pour cela, votre programme doit lire sur l'entrée standard :

1. un couple d'entiers au format : `rowstart,colstart`, représentant les coordonnées (en partant de 0) de départ de votre chemin. Bien entendu, la case correspondante ne sera jamais un mur (i.e. un #).

2. un couple d'entiers au format : `nbrow,nbcol`, représentant les dimensions du labyrinthe
3. `nbrow` lignes comportant `nbcol` caractères qui sont soit `#`, soit espace.

Attention : pour les deux premières lignes, les entiers sont bien séparées par des virgules.

Ensuite, votre programme doit lui-même trouver où se trouve la sortie puis trouver un chemin entre la position de départ et cette sortie. A noter qu'il peut y avoir plusieurs solutions à ce problème. L'essentiel est d'en trouver une.

Enfin votre programme doit écrire sur la sortie standard :

1. un entier N correspondant aux nombres de cases du chemin, **en incluant la case de départ et de sortie.**
2. N lignes au format `row,col` représentant les coordonnées des cases du chemin **dans l'ordre de parcours.**

Le tableau 1 donne un exemple d'entrée et un extrait d'une sortie possible.

entrée	sortie
5,1	26
7,13	5,1
#####	4,1
# # # # # #	3,1
# # # # # #	3,2
# # # # # #	3,3
### ### # ###	...
# #	...
#####	5,10
	5,11
	5,12

TABLE 1 – Exemple d'entrée et de sortie possible

3 les ressources

Pour vous aider dans la réalisation du programme, vous trouverez sur <http://cours-info.iut-bm.univ-fcomte.fr>

un article dans la section `hackaton` → édition 2018, portant le même titre que l'exercice. Il contient un lien permettant de télécharger un canevas de code, ainsi que le fichier d'entrée donné ci-dessus.

Bien entendu, vous êtes libres d'utiliser ou non ce canevas, mais c'est un gain de temps que de s'en servir comme base.