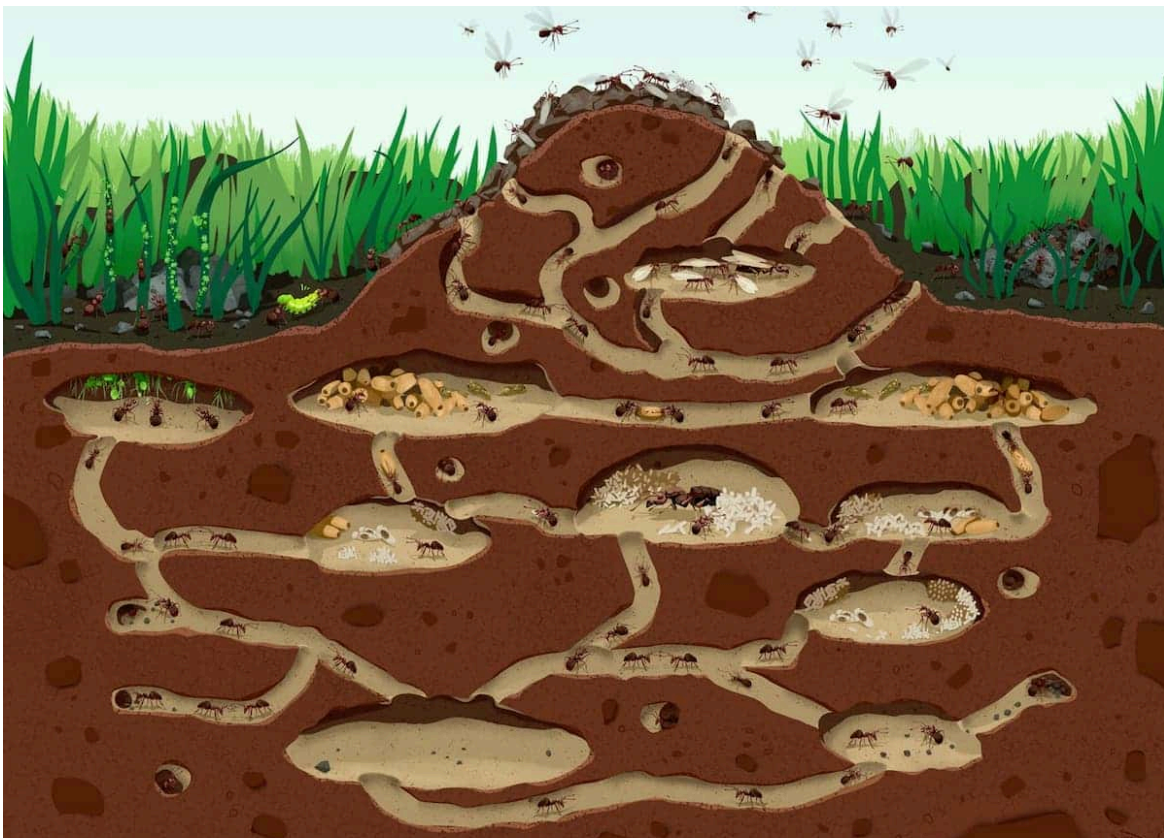


Une vie de fourmi

What do you call a 100 year old ant ? An antique !

Un trou, des tunnels et des salles



Une colonie de fourmis rouges a construit sa fourmilière sous terre, dans un vaste champ. Cette **colonie** de fourmis est composée de F fourmis, F étant un nombre entier naturel.

Une **fourmilière** est un assemblage de différentes **salles**, reliées les une aux autres par des **tunnels**.



Les deux salles les plus vastes de la fourmilière sont le **vestibule** et le **dortoir**. La première est située à l'entrée de la fourmilière et la seconde est située au fond. Celles-ci sont respectivement notés S_v et S_d .

Les autres salles de la fourmilière sont variables en **nombre** et en **taille**. Celles-ci sont notées $S_1, S_2, S_3, \dots, S_N$. Une salle notée $S_N\{X\}$ est une salle pouvant accueillir X fourmis simultanément. Un tunnel reliant deux salles est noté ainsi :

$$\rightarrow S_v - S_1$$

$$\rightarrow S_v - S_5$$

$$\rightarrow S_5 - S_d$$

L'exemple ci-dessus informe qu'il existe :

- un tunnel reliant le **vestibule** de la fourmilière à la **salle 1** (S_1)
- un tunnel reliant le **vestibule** de la fourmilière à la **salle 5** (S_5)
- un tunnel reliant la **salle 5** (S_5) de la fourmilière au **dortoir** (S_d)

Au lit, vite et bien

A la tombée de la nuit, tous les fourmis se retrouvent dans le vestibule de la fourmilière, se racontent leur journée, et se préparent à rejoindre le dortoir pour se reposer. Le vestibule étant près de la surface, n'est pas un endroit très sûr. L'ensemble des fourmis doit donc rejoindre le dortoir le plus rapidement possible afin d'éviter les prédateurs. Cependant, il faudra considérer les éléments suivants :

- Les fourmis se déplacent toutes à la **même vitesse**.



- Les salles de la fourmilière ne peuvent accueillir qu'**une seule fourmi** à la fois (en dehors du vestibule et du dortoir) sauf si **spécifié autrement**.
- Une fourmi ne peut s'engager dans un tunnel que si la **salle** de destination est **vide** (en dehors du dortoir) ou si la fourmi qui l'occupe est en **train de partir**.
- Les tunnels sont traversés **instantanément** par les fourmis (c'est plutôt des portes, en fait).
- L'intégralité des fourmis doivent **rejoindre** le dortoir en un **minimum d'étapes**.

Lors d'une étape, chaque fourmis peut **attendre** dans la salle où elle se trouve ou se déplacer dans une **salle adjacente**. Lorsque la fourmi f_1 se déplace du vestibule à la salle 1, on note alors : $f_1 - S_v - S_1$. Il n'est pas nécessaire de noter qu'une fourmi attend.

Considérons l'exemple d'une **étape** E_1 se présentant comme suit :

+++ E_1 +++

$f_1 - S_v - S_1$

$f_2 - S_v - S_2$

Ici, deux fourmis quittent le vestibule : l'une vers la **salle 1** et l'autre vers la **salle 2**.



Cas simple

Soit une **fourmilière** abritant 3 fourmis dont les salles (S_v , S_1 , S_2 et S_d) sont reliées entre elles de la manière suivante :

$$S_v - S_1$$

$$S_v - S_2$$

$$S_1 - S_d$$

$$S_2 - S_d$$

Une solution pour déplacer les 3 fourmis au dortoir serait comme suit :

$$+++ E_1 +++$$

$$f_1 - S_v - S_1$$

$$f_2 - S_v - S_2$$

$$+++ E_2 +++$$

$$f_1 - S_1 - S_d$$

$$f_2 - S_2 - S_d$$

$$f_3 - S_v - S_1$$

$$+++ E_3 +++$$

$$f_3 - S_1 - S_d$$

Les fourmis ont donc rejoint le dortoir en 3 étapes.



Ceci n'est pas un exercice

Pour chaque fourmilière proposée ([à récupérer ici](#)), vous devrez :

- représenter le graph correspondant à chaque fourmilière (schéma)
- afficher l'ensemble des étapes nécessaires au déplacement des fourmis, comme montré dans l'exemple précédent.



Compétences visées

- Algorithmique

Aller plus loin

- Créer une petite interface simple pour visualiser le cheminement des fourmis à l'intérieur de la fourmilière
- Réaliser un benchmark des différents algorithmes connus



Rendu

L'évaluation de ce projet se fera sur deux aspects :

1. Une présentation explicative de votre travail sous forme de diapositives.
2. Un repository github public nommé **uneviedefourmi**, contenant les éléments suivants :
 - a. Un fichier **ants.cpp & ants.hpp** avec les différentes classes et fonctions permettant le bon déplacement des fourmis au sein de la fourmilière.
 - b. Un fichier **main.cpp** de la résolution des fourmilières.
 - c. Un fichier **README.md** expliquant la problématique, les solutions apportées et une conclusion de votre travail.

Base de connaissances

- [Matrice d'adjacence](#) : Adjacency matrix - Graph Theory Tutorial.
- [Weighted Graphs \(and graphs in general\) - C++ Forum](#)
- [Graph in C++. All About Graphs | by Parul chaddha | Medium](#)
- [Tutorials for SFML 3.0](#)
- [SDL library in C/C++ with examples - GeeksforGeeks](#)