

# ~ Rapport Projet C ~

## Robot Pompier



Evan Guélard - CIR 1 - 2013/2014

## **Sommaire :**

1/ Introduction .....	3
2/ Algorithme utilisé .....	4
3/ Difficultés rencontrées .....	5
4/ Conclusion .....	6

## 1/ Introduction

Ce projet a pour but de programmer en langage C une simulation de déplacement d'un robot qui doit éteindre un feu dont la position est inconnue. Pour cela le robot doit déjà se rendre jusqu'à l'extincteur. Il connaît le plan de l'appartement, la position de l'extincteur, mais pas celle du feu. Une fois le robot en possession de l'extincteur, il doit trouver le feu et se rendre sur une de ses extrémités pour l'éteindre entièrement.

Pour cela, le programme doit ouvrir le plan de l'appartement. Sur la première ligne du plan est écrite la taille de l'appartement. Sur ce plan, les murs sont noté par un 'x', l'extincteur par un 'E', la position de départ du robot par un 'D' et les positions du feu sont numérotées.

Le robot doit trouver le chemin le plus court pour aller à l'extincteur, ensuite il doit parcourir tout l'appartement pour trouver le feu et l'éteindre.

## 2/ Algorithme utilisé

Pour trouver le **chemin le plus court** pour aller d'un point à un autre avec possibilité d'obstacles, j'ai étudié plusieurs algorithmes, comme 'Dijkstra' et 'A\*', mais j'ai eu un peu de mal à les comprendre. J'ai donc cherché plus d'informations sur celui de Dijkstra et pendant mes recherches, j'ai discuté avec un collègue de classe qui a évoqué l'algorithme 'Sample' et qui m'a expliqué les bases de cet algorithme. J'ai donc fait des recherches sur cet algorithme ; je l'ai assez vite compris et surtout j'ai réussi à visualiser comment le mettre en place pour le projet.

L'algorithme 'Sample' fonctionne de la façon suivante :

Soit 2 Points, D (départ) et A (arrivée), on prend le point A et on lui donne ce que j'ai appelé un « poids » qui correspond aux nombres de déplacements à faire pour arriver au point voulu, le poids de A est 0 car nous sommes au point voulu.

Ensuite pour chaque point adjacent à A, on donne un poids de 1, car il faut 1 pas pour arriver à A. Après pour chacun des points

adjacents on leur donne un poids qui est supérieur de 1 par rapport à la case de référence. Les case auront des poids, l'algorithme continue jusqu'à ce que les coordonnées d'une des case corresponde au coordonnées de la case de Départ (car nous avons commencé l'algorithme à la case d'arrivée).

Une fois la case de départ ayant son poids, il suffit de faire une fonction qui pour la case actuelle regarde les poids des cases adjacente et se déplace sur la case ayant le poids le plus petit, ainsi nous arrivons à la case d'arrivée par un des chemins les plus rapides.

```

X X X X X X X X X
X      X X  X  X
X  X  X      X
X  X X      X  X
X  X      X  X
X A      X X  X  X
X  X      X  X  X
X  X X      X  X
X      X      X
X X X D X X X X X

```

On part de A

```

X X X X X X X X X
X 4 5 6 X X  X  X
X 3 X  6 X      X
X 2 X X 5 6  X  X
X 1 X 3 4 X      X
X A 1 2 X X  X  X
X 1 X 3 4 X  X  X
X 2 X X 5 6  X  X
X 3 4 5 6 X      X
X X X D X X X X X

```

L'algorithme "place" les 1 ensuite les 2 ... tant qu'il n'est pas arrivé à D

A correspond à 0 et D à 6

### 3/ Difficultés rencontrées

J'ai eu du mal à trouver l'algorithme du chemin le plus court qui me convenait, ce qui m'a fait perdre du temps. Ensuite j'ai eu pas mal de problèmes du type « erreur de segmentation » et cela m'a pris du temps pour les résoudre.

Donc, au final j'ai eu un léger manque de temps et une petite baisse de moral à la fin du projet. Je n'ai pas fait la SDL pour privilégier les commentaires et ne pas être pris de court au dernier moment si elle ne fonctionnait pas. Le minimum que je m'étais fixé était que mon robot éteigne le feu avec un algorithme de recherche de bonne qualité (le moins de pas possible), mais ce minimum je ne l'ai obtenu sans trop de bugs ni d'erreurs que la veille de rendre le projet ; j'ai donc douté sur le fait de réussir la SDL dans les temps, j'ai donc préféré ne pas commencer la SDL et bien commenter mon code plutôt que de prendre le risque que la SDL ne marche et que mon code reste sans commentaire.

Au dernier moment j'ai voulu commencer une version colorée du programme dans la console, mais le stress a pris le dessus et je n'ai pas terminé ni ajouté au programme.

De plus, par manque de temps et d'idées je n'ai pas fait l'algorithme de recherche du feu. J'ai hésité à faire de l'aléatoire pour trouver le feu si on n'est pas encore passé dedans mais je me suis dit que ça ne servait pas à grand-chose, et encore moins si la carte est de grande taille. Donc j'ai laissé comme tel.

Sinon, au moment de l'évaluation du passage de la seconde carte, mon robot est mort et je pense savoir pourquoi. En effet il y avait un bug, qui faisait que quand le robot terminait un déplacement, il redescendait d'une case. Du coup, quand mon robot a pris l'extincteur et est retourné dans le feu pour l'éteindre ; après avoir fini son déplacement, il a éteint le feu, mais comme il est descendu d'une case avant de l'avoir éteint, il se retrouve dans un feu '2' alors qu'il ne devrait pas et par conséquent il meurt. Je ne considère pas ça totalement comme un échec car il repasse dans le feu avec l'extincteur et sans ce bug, il aurait dû l'éteindre avant de redescendre d'une case et mourir.

## 4/ Conclusion

J'ai eu certains problèmes et je regrette de ne pas avoir eu le temps de faire la SDL car j'avais une bonne idée de design. Mais je suis quand même content de mon travail, je ne me pensais pas être capable de travailler autant le soir et à chaque moment de la journée où j'avais le temps sur ce projet comme je l'ai fait.

Au final mon programme trouve l'extincteur avec un chemin rapide, et s'il est passé sur le feu, il retourne l'éteindre, sinon il s'arrête.

Pour conclure, je vais parler des améliorations du projet auxquelles j'ai pensé. Tout d'abord, le projet me semble un peu long. Si nous savions exactement où le feu se situe, je pense que ce serait mieux niveau temps. Ensuite, je pense que des séances réparties sur plus de jours pourraient aussi aider, car on pourrait plus avancer chez nous. Pour finir, un professeur de plus lors des séances du projet pourrait aider à aller plus vite au niveau de l'aide personnelle. Pour ma part ça a été mais certains devaient attendre la fin de la séance pour être aidés, et perdaient un peu de temps du coup.