GALASSI Laurent

03/04/2016

ARNOLD Julien

GOURDEAU Simon

**Rapport**

**Projet Tram**

Faculté de Sciences et Techniques



03/04/2016

Sommaire

## I) Introduction

### Présentation

### Fonctionnalités

## II) Conception

### Déroulement du projet

### Technologies utilisées

### Problèmes rencontrés

## III) Conclusion

I.1) Présentation

Le projet de gestion d’un réseau de tramways a pour but de simuler un trafic représenté de manière graphique dans une application développée en C++.

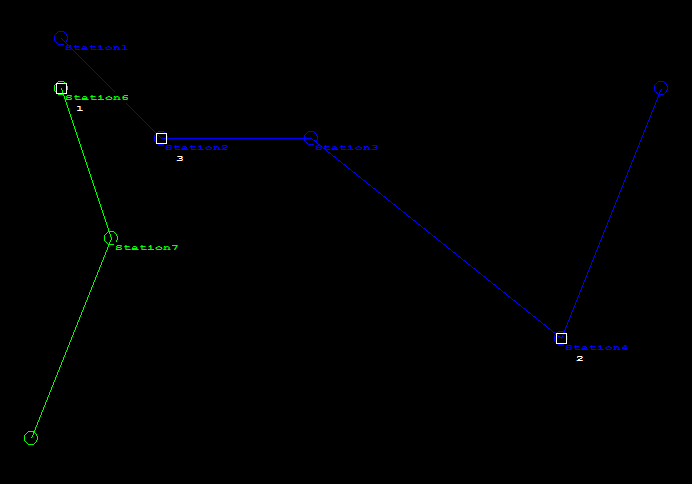
Les données qui sont nécessaires au fonctionnement de cette simulation doivent être stockées dans un fichier afin d’initialiser correctement le programme.

Les lignes composées de stations doivent êtres parcourues par des trams dans les deux sens qui respectent leur distance de sécurité.

I.2) Fonctionnalités

L’utilisateur peut dans un premier temps choisir plusieurs options qui s’affichent à lui.

Une fois la simulation lancée, la simulation est représentée de la manière suivante :



* Les cercles représentent les station, accompagnées de leur nom
* Les carrés les trams, accompagnés aussi de leur nom
* Les lignes sont différenciées par différentes couleurs

Les trams se déplacent correctement sur les lignes, s’arrêtant à leurs station en respectant leur distance de sécurité.

II.1) Déroulement du projet

Dans un premier temps nous avons organisé une réunion afin de réfléchir à la gestion du projet et à la structure globale des données afin de pouvoir les représenter correctement par la suite.

Une fois avoir réuni les idées afin d’implanter une structure cohérente, nous avons implanté une boucle temporelle qui permet de simuler pendant une durée entrée par l’utilisateur.

Nous avons réparti le travail restant entre nous en plusieurs tâches :

* Laurent Galassi a réalisé les calculs de distance et de parcours des trams
* Simon Gourdeau s’est penché sur la représentation graphique des données
* Julien Arnold a implanté le format des données à charger ainsi que leur chargement. Il a aussi créé la liste chaînée de Trams.

II.2) Technologies utilisées

Dev C++ :

Nous avons choisi cet EDI car nous l’avons utilisé au cours de l’année, il est performant et facile d’utilisation.

La bibliothèque Winbgi :

Elle permet le dessin et l’affichage de lignes, carrés ou cercle ainsi que d’afficher du texte. Elle n’est pas compliquée à mettre en place c’est pour cela que nous l’avons choisie.

La bibliothèque Tinyxml :

Julien Arnold a choisi le format XML car il correspond parfaitement aux demandes du projet qui sont un fichier facilement modifiable et chargeable.

Cette bibliothèque permet donc de manipuler ce format de données en les parcourant nœud par nœud. Le XML étant un format standardisé, cela permet une compréhension simplifiée de son fonctionnement.

Github :

Nous avons donc choisi d’utiliser Github afin de pouvoir plus facilement travailler ensemble sur le projet. C’est un logiciel très répandu qui permet d’avoir une gestion de version du code. Chaque personne peut envoyer ses modifications via des « commit » auxquels il donne un titre et une description pour que les autres membres du projet puissent savoir ce qu’il a fait.

Skype :

Ce logiciel de conversation nous a permis de discuter du projet à distance et de nous réunir.

II.3) Problèmes rencontrés

Notre répartition du travail a dû être revue car un membre a abandonné le projet.

Lorsque nous avons implémenté la boucle temporelle, nous nous sommes trompé dans le type de variable, à la place d’un clock\_t, nous l’avions défini en int, ce qui a faussé les calculs.

Pour parcourir le fichier XML, il faut correctement boucler entre les différents nœuds, mais pour accéder à une station, il faut accéder au-sous nœud de ligne, ce que Julien avait négligé. Il y avait donc une erreur d’accès à l’élément et pour régler ce problème il suffisait de pointer sur « l’enfant » de cet élément.

Lors de l’affichage, les dessins sur la fenêtre clignotaient à cause de la boucle qui les efface et les redessine à chaque itération. Pour pallier ce problème nous avons ajouté un Sleep(1000) qui fait une pause à chaque seconde.

Un seul tram ne s’arrête pas correctement durant les temps d’arrêt, mais cela ne se produit que certaines fois durant la simulation. Laurent Galassi n’a pas réussi à régler ce problème.

Après avoir mis un settextstyle pour améliorer la visibilité du texte sur l’interface graphique, un bug est apparu si l’on voulait relancer la simulation plusieurs fois de suite. La durée restante en seconde ne s’affiche plus que lors de la première simulation. Dans les suivantes, elle n’apparaît que lorsque la valeur a atteint 0. Après avoir retiré tout nos settextstyle, ce bug est resté présent sans aucune raison.

III) Conclusion

Ce projet nous a permis de mettre en pratique les pointeurs dans une situation réelle.

Le travail en groupe était essentiel et une mauvaise organisation aurait causé l’échec de sa réalisation.

Les difficultés rencontrées nous ont inciter à chercher de nous-même comment les résoudre plutôt que de les laisser sans solution.

Nous vous remercions pour l’aide apportée lors de notre réunion grâce à laquelle nous avons acquis une meilleure compréhension de certains points.