

**POLYTECHNIQUE
MONTRÉAL**

UNIVERSITÉ
D'INGÉNIERIE



Paul Clas, matricule : 1846912

Mazigh Ouanes, matricule : 1721035

Rapport du TP2 : Automates

LOG2810 Structures Discrètes

Présenté à :

John Mullins

Justine Pepin

Paulina Stevia Nouwou Mindom

Remis le

3 Avril 2019

Département de génie informatique et génie logiciel

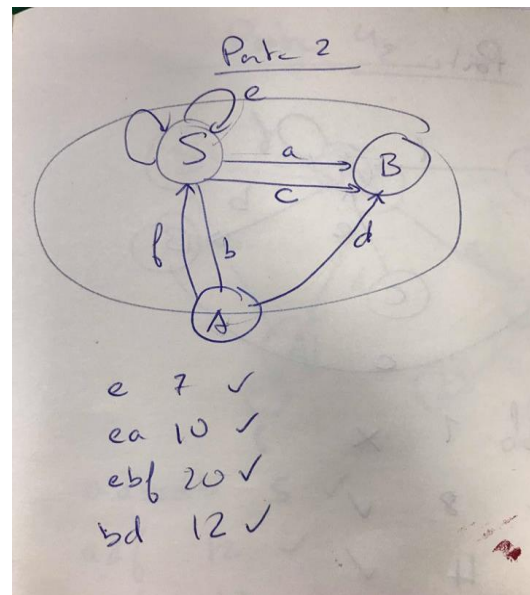
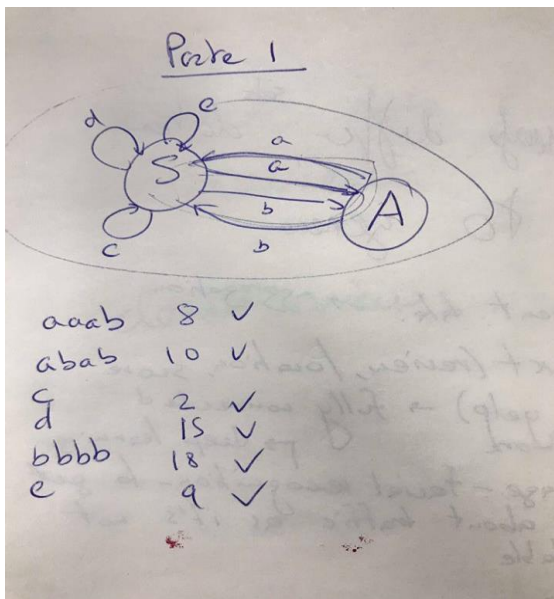
Polytechnique Montréal

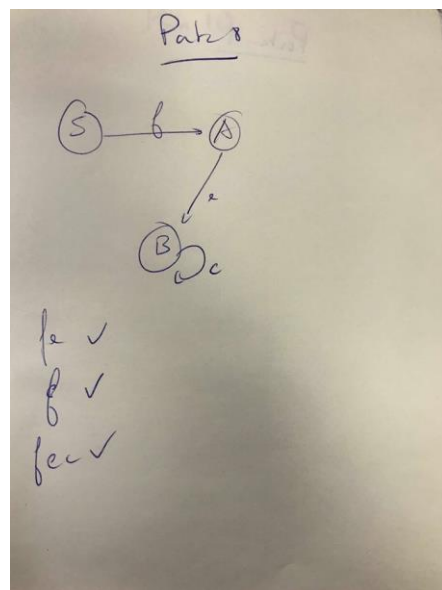
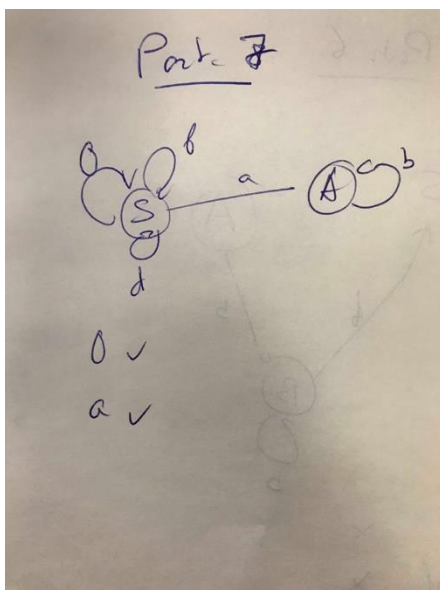
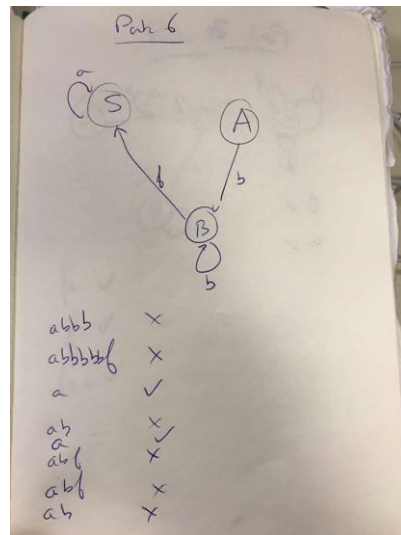
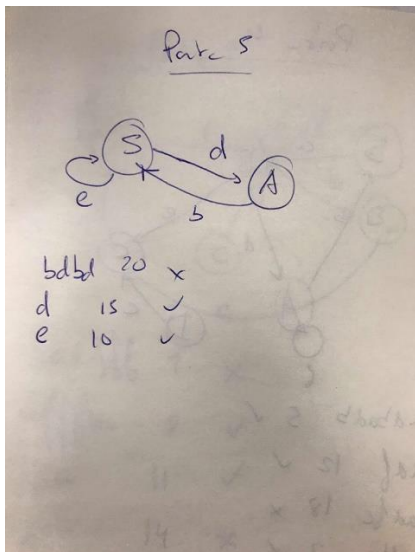
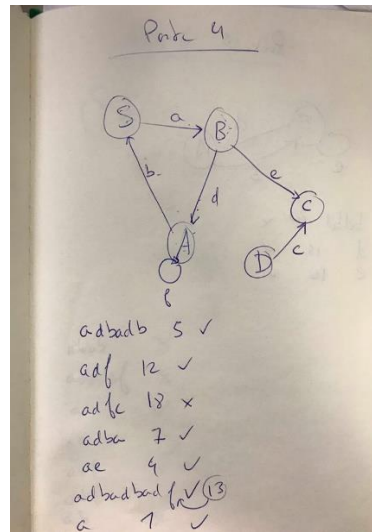
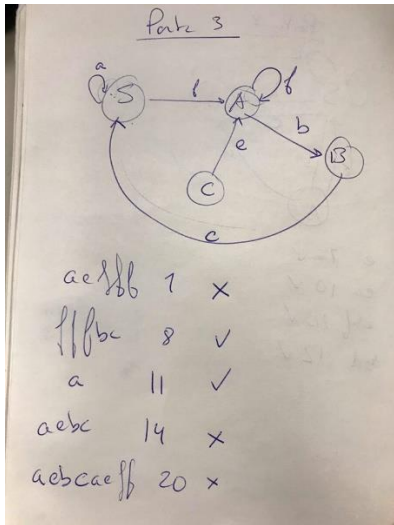
Introduction

L'objectif de ce laboratoire est de se familiariser avec les concepts des graphes, les automates ainsi que la génération de langage. En utilisant les concepts énumérés précédemment, nous devons concevoir une version simplifiée du jeu populaire : « Dont vous êtes le héros ». Ce jeu est assez simple, il permet au joueur de choisir une série de portes afin de pouvoir venir à bout d'un « boss » avec l'intermédiaire de mot de passe qui sont accumulés dès la première porte. Nous ne sommes rapidement familiarisés avec le fonctionnement de ce jeu en faisant une recherche en ligne.

Présentation de nos travaux

Avant tout, pour ce deuxième travail pratique, nous avons décidé de coder nos fichiers en java puisque ce langage est assez familier et que nous l'utilisons dans un autre cours. Lors de notre phase de conception, nous avons commencé par essayer de comprendre comment les automates sont inter-reliées entre elles d'un point de vue graphique. Les images ci-dessous montrent certains schémas que nous avons élaborés pour une meilleure compréhension du travail. Une bonne compréhension de la logique de l'énoncé est primordiale afin de pouvoir commencer à développer le travail désiré, avec toutes ses contraintes, sur eclipse en langage de programmation java.





Par la suite nous avons séparé les différentes tâches présentées dans l'énoncé de ce travail pratique que nous devons effectuer en six classes : main.java, instruction.java, password.java, labyrinthe.java, porte.java et arrayList.java. Tous ces fichiers vous ont été remis.

De plus, il est important de vous mentionner que nous avons changé la structure des fichiers .txt fournis avec l'énoncé. Merci de bien vouloir utiliser les fichiers .txt fournis avec notre remise.

Voici quelques brèves descriptions des classes que nous avons implémentées.

main.java : le main sert à afficher l'interface du jeu en donnant plusieurs options à l'utilisateur: entrer dans le labyrinthe, afficher le chemin parcouru ou quitter ouvrir une porte.

instruction.java : ici se trouvent plusieurs méthodes d'affichage et de contrôles des automates.

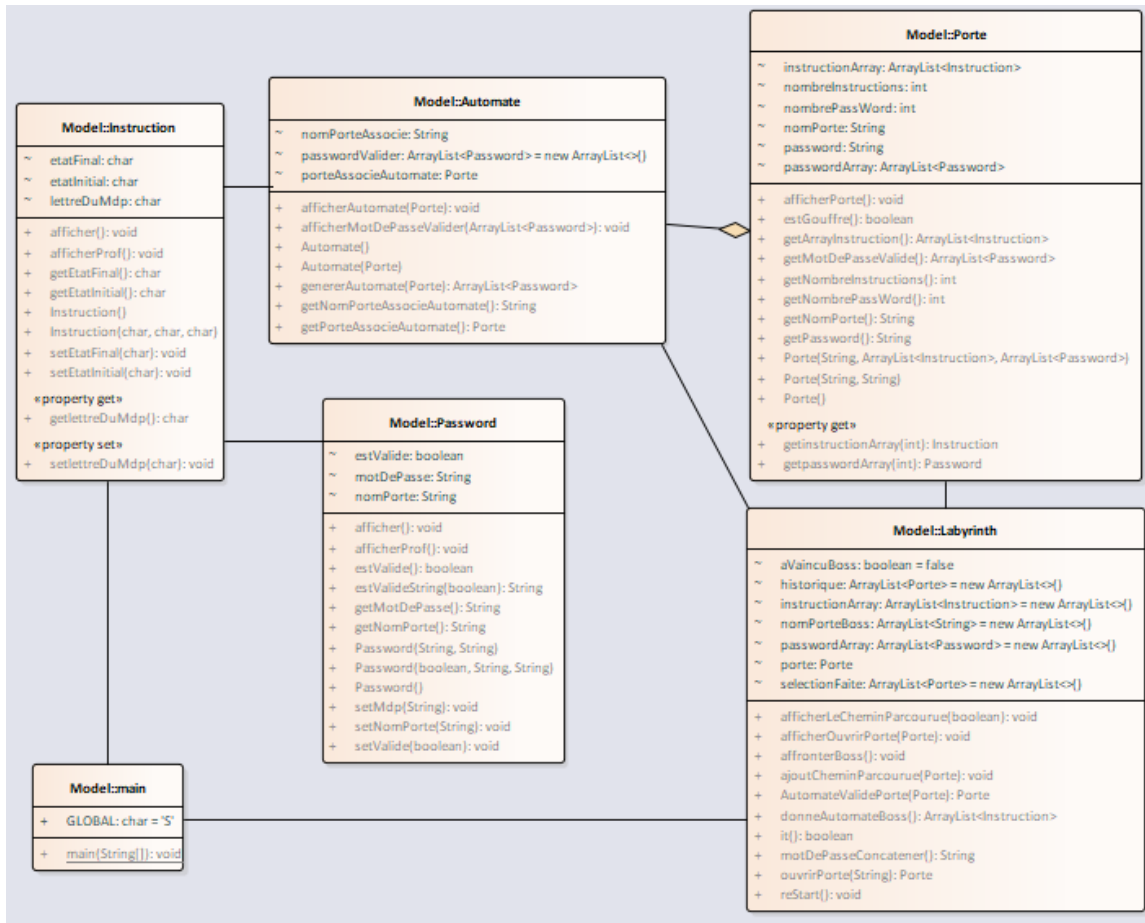
password.java : Cette classe permet de gérer tout ce qui est relatif au mot de passe dans le tp.

labyrinthe.java : Permet de lire les fichiers textes et d'afficher les chemins encourus. Plusieurs méthodes de gestion du jeu se trouvent ici.

porte.java : Permet de gérer tout ce qui est portes.

automate.java : Permet de générer et créer des automates. Elle possède tout d'abord la fonction genererAutomate() qui prend les règles présentes dans la porte pour créer le graphe.

Diagramme de Classe



Difficultés rencontrées

Lors du processus de conception et de développement, certaines problématiques ont fait surface quant à l'élaboration du programme. Tout d'abord, dès la conception, réaliser l'architecture du logiciel était une tâche qui a été ardue. Premièrement, nous avons dû changer la structure de nos fichiers .txt afin de pouvoir les lire convenablement sur eclipse. De plus, nous avons dû changer de structures de données à quelques reprises afin d'assurer une bonne cohésion en équipe. Néanmoins, cela nous a fait perdre beaucoup de temps. Finalement, quelques-uns des points que nous devions implémenter étaient assez difficiles. Par exemple, il nous était facile de se perdre dans nos index lorsque nous avions

implémenter certaines fonctions dans le labyrinthe.java afin de pouvoir trouver le chemin menant vers le boss.

Conclusion

En guise de conclusion, la conception du jeu « dont vous êtes le héros » qui utilise des automates ainsi que la théorie du langage pour créer des mots de passe nous a permis de se familiariser avec le principe d'utilisation de graphe pour un but de validation de mots de passe ainsi que les règles de création de langage.

a. Temps passé sur le TD

1 semaine : Mercredi 9h, Jeudi 6h, Vendredi 6h, Samedi 4h, Dimanche 7h Lundi 6h, Mardi, 6h,

Total : environ 40h

b. Bibliographie

Edwin Grappin, Spatial Segregation in cities, 15 Novembre 2012,

<http://probaperception.blogspot.com/2012/09/spatial-segregation-in-cities.html> Consulté le 25 février.

Edwin Grappin, The best Strategy to play Guess Who, 25 Septembre 2012,

<https://probaperception.blogspot.com/2012/10/the-best-strategy-to-play-guess-who.html> Consulté le 25 février

John Mullins, Note de cours LOG2810, Session Hiver 2019, Polytechnique Montréal