Rapport sur le Travail #1

Présenté à Adam Joly

Effectué le 20 octobre

À Trois-Rivières

Jérémy Veillette étudiant en

Baccalauréat en Informatique

Année 2022

À l’Université du Québec à Trois-Rivières

Text

Description automatically generated

**------**

# Rôle :

Lors de ce travail, la division des tâches n’était que futile, puisque j’ai effectué le travail seul, on peut donc considérer que l’ensemble des tâches ont été faite par moi, que ce soit l’écriture du rapport, le développement du programme, la correction du programme suivant les tests et l’application des tests.

# Problèmes et difficultés

Tôt dans le développement de l’application, des difficultés sont apparues, certaines mineures et simples à résoudre, mais une particulièrement difficile à cerner. Le point un dans les commentaires du projet où il est demandé « (1) Charger un fichier en spécifiant le chemin (relatif) du fichier. Vous pouvez charger un fichier par défaut au démarrage » fût la difficulté, je dirais, la plus compliqué à implémenter de façon simple. Finalement, j’ai réussi à établir quelque chose d’assez familier pour un utilisateur en utilisant la classe « OpenFileDialog » de C# .NET. Ceci me permit de permettre à l’utilisateur de choisir le fichier qu’il désir lire, et en appliquant un filtre pour n’avoir que les fichiers .txt dans la sélection, il n’est pas possible pour le client de choisir un mauvais type de fichier, à partir d’une page d’explorateur de fichier. Un autre problème que j’ai rencontré fut l’addition de transition dans la liste de transition des états. Lorsque j’ai codé cette action au départ, tout semblait fonctionner comme il faut, il n’y avait pas d’erreur et je n’avais donc pas de raisons apparentes, sur le moment, pour faire un débogage approfondi du système. Cependant, quand j’en suis venu à faire la validation de suite de zéro et d’un, à ce moment-là, pour vérifier le bon fonctionnement de la fonction de validation, j’ai bien dû faire du débogage et j’ai remarqué que les transitions étaient ajoutées deux fois à chaque état. J’ai, donc, procédé à un débogage plus détaillé pour, finalement, trouver que l’ajout se faisait bien à deux moments, car l’ajout se faisait dans la fonction « EvaluateTransition », mais la fonction était aussi utilisée en tant que condition dans le « if » qui déterminait si elle pouvait être utilisé. Il me fallut, donc, déplacer l’ajout de dans la fonction à après son appel dans le « if » et utiliser ce qu’elle retourne comme élément à ajouter dans la liste de transition des états.

# Guide utilisateur

Le guide suivant démontrera comment fonctionne l’application à l’aide de captures d’écrans, d’explication sommaire et démonstration de tests effectués.

Les tests qui ont été effectués sont :

1. Un exemple de fichier d’automate conforme et son chargement initial;

2. Un exemple de fichier d’automate non conforme, puisqu’une transition pour état non existant était dans le fichier à charger, et le succès du chargement en ignorant l’action;

3. Un exemple de fichier d’automate non conforme, puisqu’une action autre que celles prévues était dans le fichier à charger, et le succès du chargement en ignorant l’action;

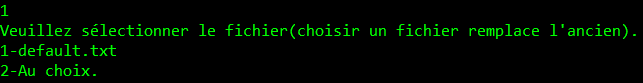
4. L’affichage de l’automate en console pour 2 automates différents avec au 4 états chacun;

5. La soumission d’un input qui retourne une acceptation dans 3 cas de tests;

6. La soumission d’un input qui retourne un rejet dans 3 cas de tests;

7. La soumission d’un input composée d’au moins un caractère qui n’est pas un 0 ou un 1, et qui retourne un rejet sans plantage.

Premièrement, comment fonctionne l’application, lorsque l’on démarre l’application, il y a un menu à quatre options. Text

Description automatically generatedComme on peut voir, il y est demandé de faire un choix parmi les quatre visible. Pour ce faire il s’suffit d’entrer le chiffre correspondant, puis d’appuyer sur « enter ». Allons-y en ordre, si l’on choisit l’option 1, il nous est donné un choix. Ici, le premier choix va charger les données correspondant à l’automate par défaut d’un dossier déjà contenu dans le projet. Le deuxième choix, cependant, donne l’option de choisir le fichier désiré à l’utilisateur. Graphical user interface, application

Description automatically generatedPour la démonstration de base nous utiliserons le fichier par défaut dans le choix que nous avions dans l’option 1. Pour l’option 2, elle sert à la liste des états et des transitions que voici. Text

Description automatically generatedDans la première colonne, les états se répètent, malgré qu’elle soit nommé « CurrentState » ou état courant, parce que certains états ont plus d’une transition, donc il paraît sensé de les énuméré à la façon ci-dessus. Ensuite, nous avons l’option, elle permet l’entrer d’une série d’uns et de zéros et la validation de celle-ci. A screenshot of a computer

Description automatically generated with medium confidenceText

Description automatically generatedFinalement, l’option 4 permet de quitter l’application. Maintenant, passons aux tests.

1. Un exemple de fichier d’automate conforme et son chargement initial;  
étape à partir du menu de base : 1->1  
Text

Description automatically generated

2. Un exemple de fichier d’automate non conforme, puisqu’une transition pour état non existant était dans le fichier à charger, et le succès du chargement en ignorant l’action;  
étape à partir du menu de base :1->2->sélectionner test\_non\_conforme\_transition.txt  
Graphical user interface, text, application

Description automatically generated

3. Un exemple de fichier d’automate non conforme, puisqu’une action autre que celles prévues était dans le fichier à charger, et le succès du chargement en ignorant l’action;  
étape à partir du menu de base : 1->2->sélectionner test\_non\_conforme\_action.txt  
Graphical user interface, text, application, email

Description automatically generated

4. L’affichage de l’automate en console pour 2 automates différents avec 4 états chacun;  
étape à partir du menu de base :1->1->2 || 1->2->sélectionner test\_deuxième\_automate.txt->2  
Text

Description automatically generatedText

Description automatically generated

5. La soumission d’un input qui retourne une acceptation dans 3 cas de tests;  
Acceptation :

pour l'automate de default.txt :

01

0001

0001101001

A screenshot of a computer

Description automatically generated with medium confidence

pour l'automate de test\_deuxieme\_automate.txt :

1

10

001011

A screenshot of a computer

Description automatically generated with medium confidence

6. La soumission d’un input qui retourne un rejet dans 3 cas de tests;   
pour l'automate de default.txt :

010

000101

010110100101

Text

Description automatically generated

pour l'automate de test\_deuxieme\_automate.txt :

11

101

001011010110

A screenshot of a computer

Description automatically generated with medium confidence

7. La soumission d’un input composée d’au moins un caractère qui n’est pas un 0 ou un 1, et qui retourne un rejet sans plantage.   
A screenshot of a computer

Description automatically generated with medium confidence