

Université de la Manouba
Ecole Nationale des Sciences de l'Informatique



Mini-rapport du Stage d'été II1
-Stage de Programmation-
Sujet

Mise en œuvre de l'évolution d'une machine de Turing

Réalisé par

Marnaoui Iheb eddine

Jouini Nesrine

1. Présentation générale du sujet

En 1937, le mathématicien Alan Turing publie un article fondateur de la science informatique: il y présente sa machine de Turing, le premier calculateur universel programmable plus connus aujourd'hui sous le nom d'ordinateurs et invente ainsi les concepts de programmation et de programme.

Les deux principaux types de machines de Turing sont les machines déterministes où pour un état donné et un symbole lu, il existe une unique action à effectuer et les machines non-déterministes où à chaque étape, la machine a le choix entre plusieurs transitions.

Une machine de Turing est utilisée soit pour calculer (pour un mot donné en entrée, la machine retourne un résultat de calcul) soit pour accepter (pour un mot donné en entrée, la machine répond par oui si ce mot est accepté et par non sinon).

Dans ce projet, nous avons créé un simulateur de machine de Turing afin d'aider les utilisateurs à mieux appréhender le fonctionnement de cette machine en leur offrant la possibilité de tester leur machine et de visualiser son exécution pas à pas. Les machines considérées sont des machines déterministes qui calcul ou qui accepte.

2. Présentation des fonctionnalités demandées de l'application

Le but de notre projet est de mettre en œuvre une application qui permet de recevoir en entrée la description d'une machine de Turing, d'un mot à analyser et de montrer graphiquement l'évolution de la machine sur les états et le ruban d'entrée.

La description saisie par l'utilisateur sera exécutée par une machine de Turing déterministe constitué d'un ruban infini.

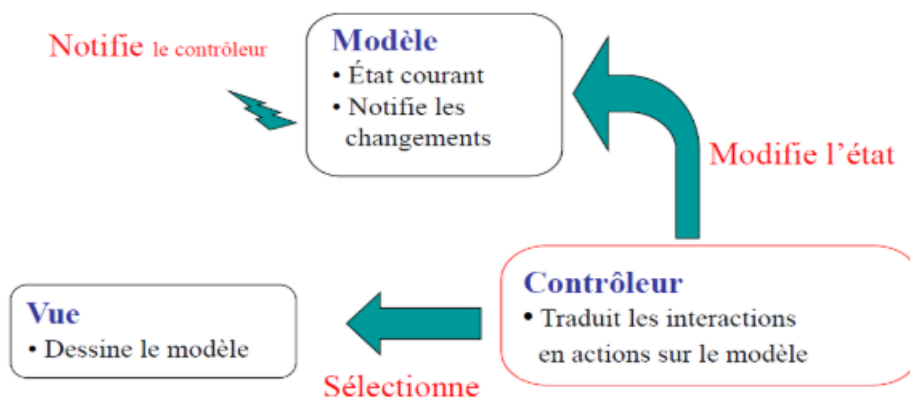
La tête de lecture écriture travaille sur la cellule de la bande située devant elle. Elle peut lire le contenu de cette case ou effacer ce contenu et y écrire un autre élément.

Il existe un dispositif d'entraînement permettant de déplacer la tête de lecture écriture d'une case vers la Droite ou vers la Gauche.

3. Présentation de la conception de la solution proposée

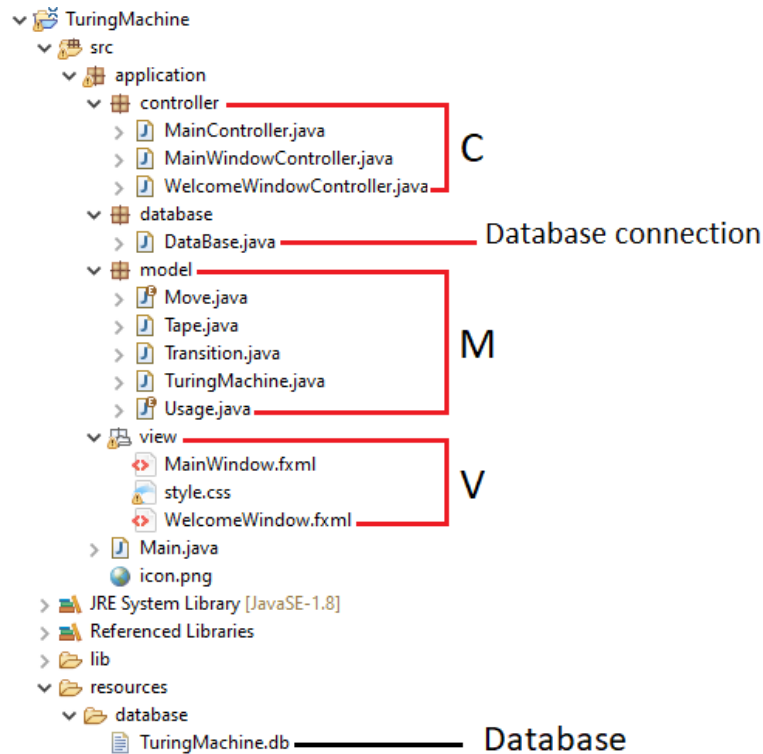
3.1. Architecture du programme :

Nous avons organisé notre code selon la structure Modèle-Vue-Contrôleur (MVC) qui est un pattern design qui permet de séparer le code en trois parties :



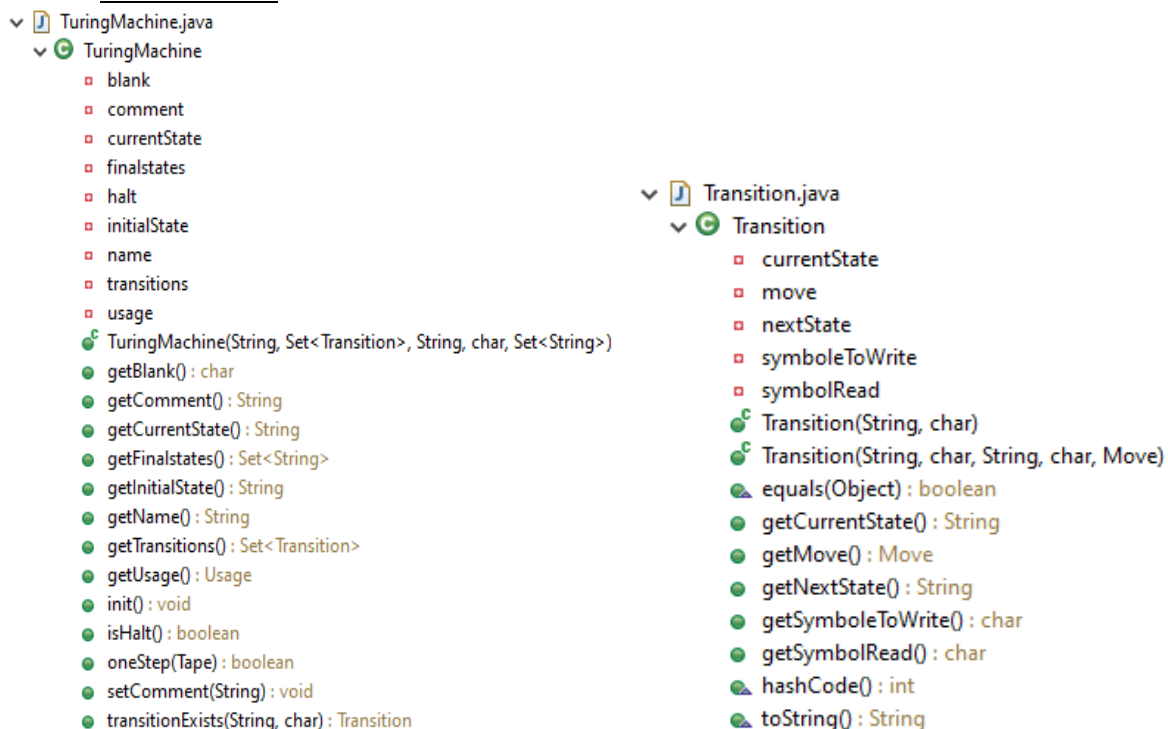
- Modèle : représente un objet transportant des données.
- Vue : représente la visualisation des données contenues dans le modèle.
- Contrôleur : interprète les actions de l'utilisateur, modifie les données des modèles(objets) et met à jour la vue.

Organisation des fichiers

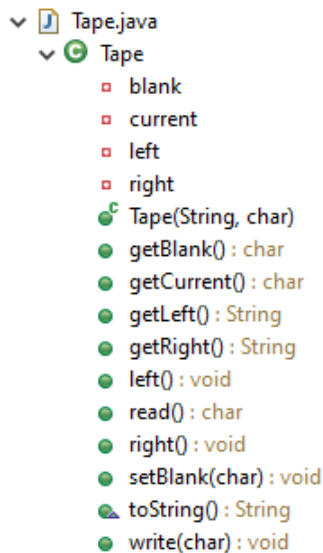


3.2. Détails de la solution

1. Les modèles :

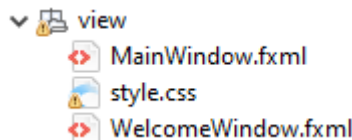


Ces deux classes sont la représentation d'une machine de Turing dans notre programme.



Cette classe représente le ruban de la machine de Turing.

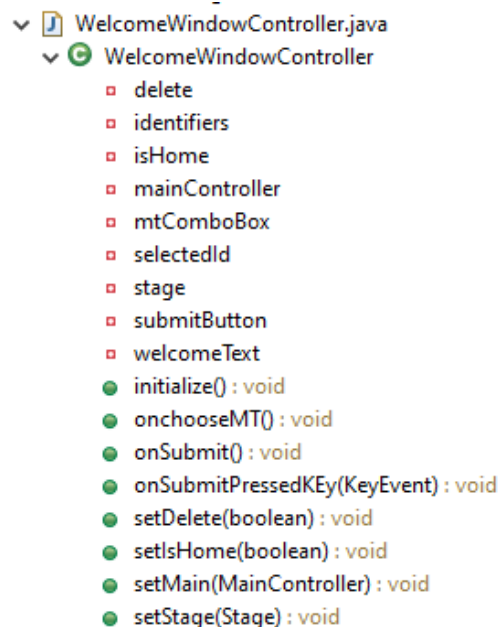
2. Les vues



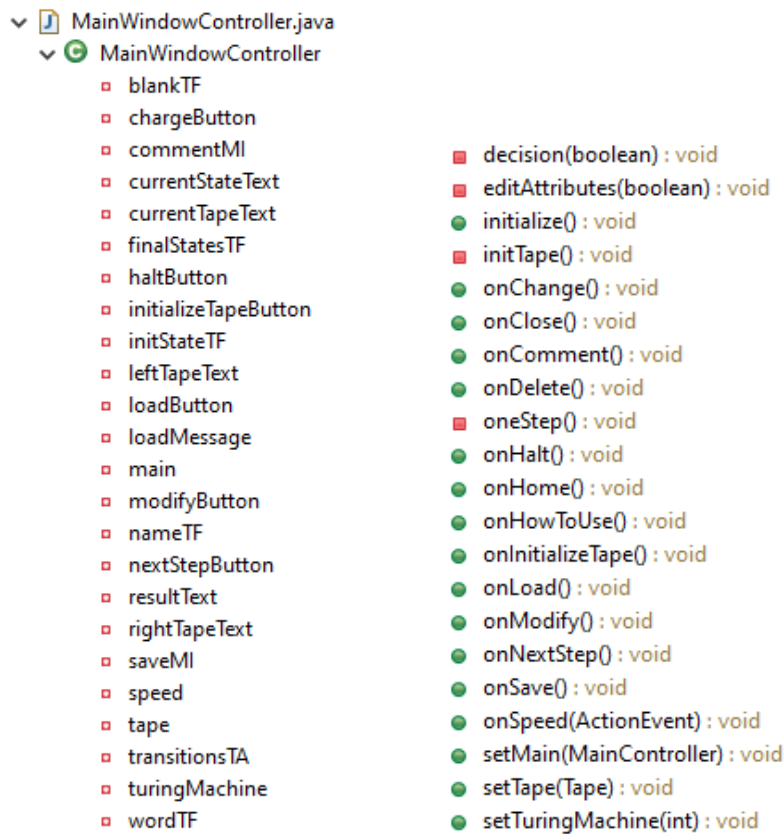
Les deux fichiers MainWindow.fxml et WelcomeWindow.fxml contiennent le code en fxml des deux interfaces graphiques qui composent notre programme.

Le fichier style.css est une feuille de style que nous utilisons pour l'interface MainWindow.

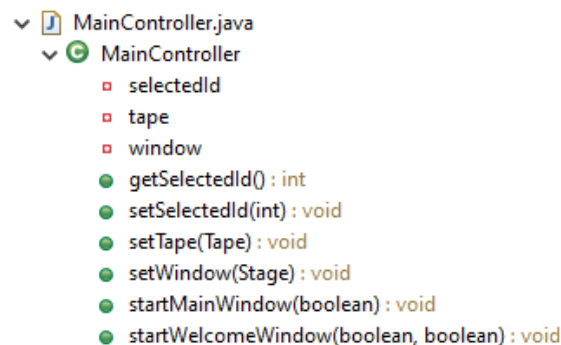
3. Les contrôleurs



Cette classe contrôleur est associée à l'interface graphique WelcomeWindow, nous y implémentons les fonctionnalités réalisées par cette interface.

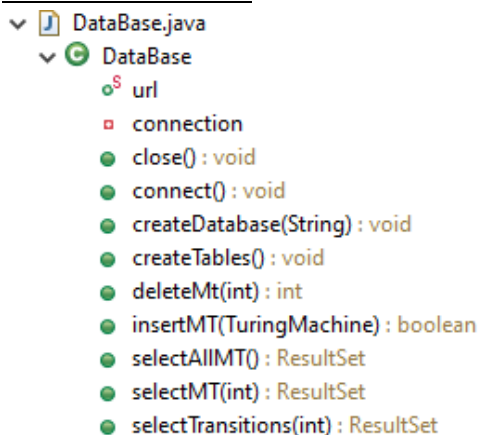


Cette classe contrôleur est associée à l'interface graphique MainWindow.



Cette classe fait la liaison entre les deux contrôleurs associés aux deux interfaces graphiques de notre application.

4. La base de données



Cette classe gère les opérations sur la base de données (création, insertion, suppression).

4. Manuel d'utilisation

4.1. Environnement logiciel

➤ Langages et bibliothèques utilisés :

→ Pour développer notre projet, nous avons opté pour le langage de programmation **JAVA** car :

- JAVA est un langage de programmation orienté objet donc notre code sera structuré et maintenable
- Il est facilement portable sur différents systèmes d'exploitation grâce à sa machine virtuelle « Write once, Run everywhere »
- La technologie Java dispose d'une bonne documentation grâce à une communauté très active



Nous utilisons la version 8 de Java Développement dans l'élaboration de notre projet.

→ Pour l'interface graphique, nous utilisons **JavaFX**. Elle est la bibliothèque de création d'interface graphique officielle du langage Java pour les applications Desktop, des applications internet riches et des applications smartphones et tablettes tactiles.

→ **SQLITE** est une bibliothèque écrite en langage C qui propose un moteur de base relationnelle accessible en langage SQL. Elle est légère, fonctionne sans serveur et peut donc être intégrée directement aux programmes. La base de données est ainsi stockée dans un fichier indépendant de la plateforme.



→ Pour connecter la base de données à notre programme, nous utilisons le **JDBC**.

La technologie **JDBC** (Java DataBase Connectivity) est une API fournie avec Java permettant de définir comment un client doit se connecter à des bases de données.

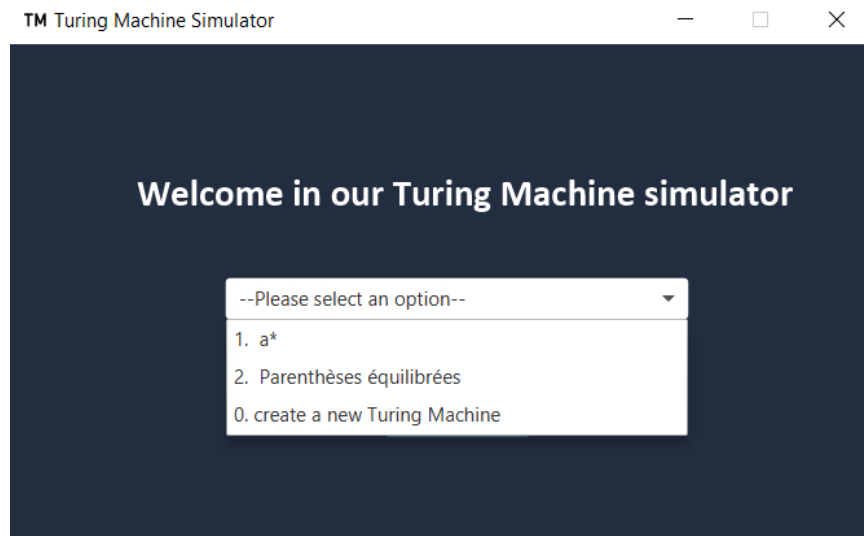
➤ Environnement de développement intégré utilisé :



Nous avons utilisé l'environnement de développement intégré ECLIPSE IDE pour développer notre projet. C'est un IDE gratuit et open-source permettant de créer des projets en JAVA en fournissant un ensemble d'outils (éditeur de texte, débogueur ...).

4.2. Présentation du travail réalisé

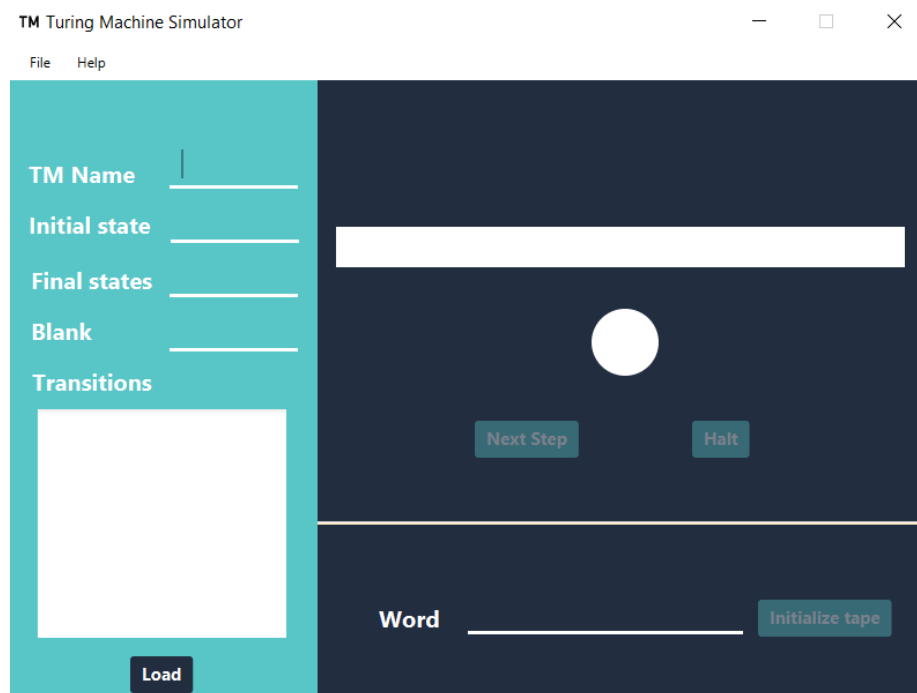
1- Menu principal :



Les options 1 et 2 sont des machines qui ont été enregistrées dans la base de données par l'utilisateur.

L'option 0 permet de créer votre propre machine de Turing.

2- On sélectionne l'option 0 :

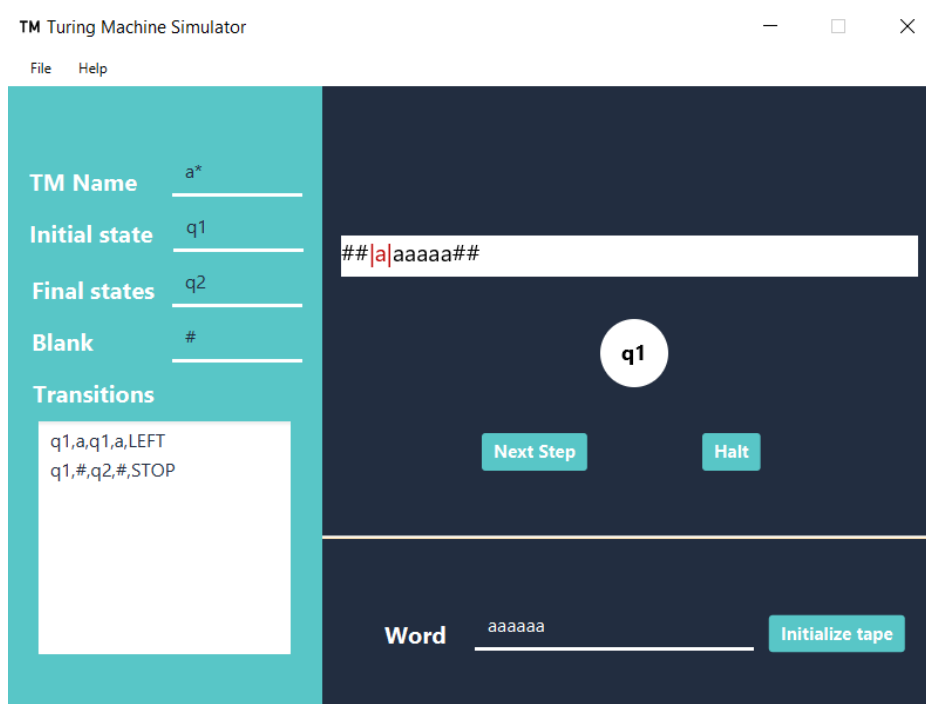


La partie à gauche représente la définition de la machine : Le nom, l'état initial, les états finaux, le symbole blanc et les transitions.

Il faut remplir les champs obligatoires puis cliquer sur le bouton « Load » pour charger la machine.

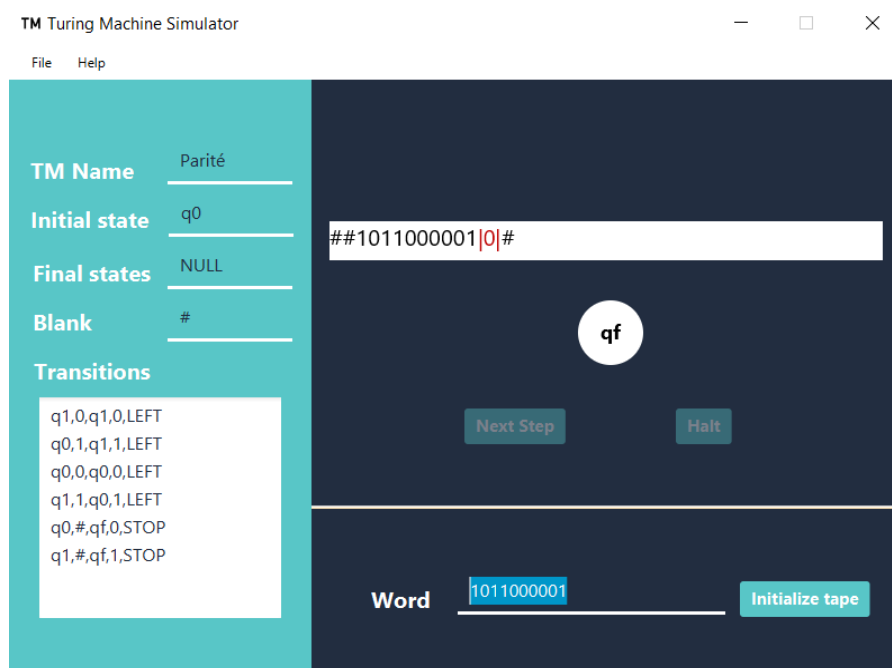
(La champ final states peut-être vide ou avoir un seul état final ou bien plusieurs, dans ce dernier cas il faut les saisir séparés par des virgules « , »).

3- Exemple d'exécution :



La partie à droite représente le ruban et son traitement par la machine présente dans la partie gauche. En écrivant le mot à traiter et en cliquant sur le bouton « Initialize tape », le mot est chargé sur le ruban. Maintenant, vous pouvez soit exécuter la machine étape par étape en cliquant sur le bouton « Next Step » ou bien en une seule fois avec le bouton « Halt ».

4- On sélectionne l'option 1 :



Dans le cas où vous demandez une machine enregistrée précédemment dans la base de données, la machine en question sera affichée dans la partie gauche et vous pourrez alors utiliser la partie droite comme précédemment décrit (paragraphe 4-Exemple d'exécution).

5- Exemple d'exécution :

Cas de mot accepté :

The screenshot shows the TM Turing Machine Simulator interface. On the left, the configuration is as follows:

- TM Name: a*
- Initial state: q1
- Final states: q2
- Blank: #
- Transitions:
 - q1,a,q1,a,LEFT
 - q1,#,q2,#,STOP

The main area displays the tape content: ##aaaaaa|#|#. The current state is q2, indicated by a circle labeled q2. The text "Word accepted" is shown in green. Below the tape, the word "aaaaaa" is entered in the "Word" field, and the "Initialize tape" button is visible. The "Next Step" and "Halt" buttons are also present.

A* est une machine de Turing qui accepte (présence d'un état final). En cliquant sur « Next Step » jusqu'à ce que la machine s'arrête (exécution étape par étape) ou « halt », la machine s'arrête et l'état final est atteint, le mot est donc accepté.

Cas de mot rejeté :

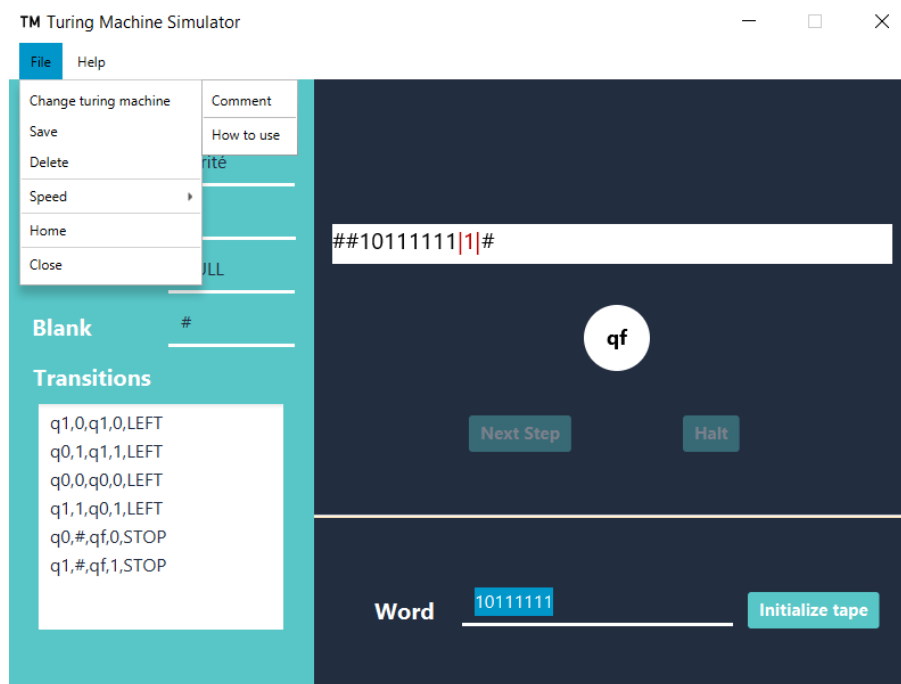
The screenshot shows the TM Turing Machine Simulator interface. On the left, the configuration is as follows:

- TM Name: a*
- Initial state: q1
- Final states: q2
- Blank: #
- Transitions:
 - q1,a,q1,a,LEFT
 - q1,#,q2,#,STOP

The main area displays the tape content: ##aaa|b|aaaaaa##. The current state is q1, indicated by a circle labeled q1. The text "Word rejected" is shown in red. Below the tape, the word "aaabaaaaaa" is entered in the "Word" field, and the "Initialize tape" button is visible. The "Next Step" and "Halt" buttons are also present.

A* est une machine de Turing qui accepte (présence d'un état final). En exécutant la machine, la machine se bloque mais l'état final n'est pas atteint, le mot est donc rejeté.

6- Autres options :



Plusieurs options sont disponibles sur notre programme.

→ Avec le menu « File »

- Vous pouvez changer la machine de Turing en cours d'exécution sans quitter le programme ce qui vous permet d'exécuter plusieurs machines sur un même ruban « Change Turing machine ».
- Vous pouvez enregistrer la machine en cours dans la base de données « Save ».
- Vous pouvez supprimer une machine existante de la base « Delete ».
- Vous pouvez contrôler la vitesse d'évolution sur le ruban (lente, moyenne, rapide) « Speed ».
- Vous pouvez revenir à l'écran d'accueil « Home ».
- Vous pouvez quitter le programme « Exit ».

→ Avec le menu « Help »

- Vous pouvez consulter le commentaire de la machine présente dans la partie gauche « Comment ».
- Vous pouvez ouvrir le manuel d'utilisation « How to use ».

5. Les problèmes rencontrés et leurs solutions

Lors de la réalisation de ce projet, nous avons rencontrés diverses difficultés que nous résumons dans le tableau ci-après.

Problèmes rencontrés	Solutions proposées
Difficulté dans la coordination et division des tâches.	Faire des réunions courtes de 5-10 minutes avant chaque séance de travail.
La base de données classique indépendante de l'application développée (MYSQL).	Utilisation d'une base de données embarquée dans l'application (SQLITE).
Développer le code en suivant le design pattern MVC. Création de l'interface graphique avec JavaFX.	Prévoir des périodes d'apprentissage avant d'appliquer le MVC ou JavaFX à notre projet.

6. Références

- <https://www.tutorialspoint.com/jdbc/index.htm> consulté le 10/09/2020
- <https://sqlite.org> consulté le 21/09/2020
- <https://openjfx.io> consulté le 10/09/2020
- https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm consulté le 21/09/2020