Project 3 Report Umut Oskay

**Simulation of a Turing Machine**

**Abstract**

In this Project we will be simulating a Turing Machine. It will be a console application. 3 classes and 1 enum are included in this Project. Classes are Main, Turing, and Transition classes. Enum is Direction.

**Turing Class**

**Text

Description automatically generated**

In Turing Class we have head, tape, transitions, state, rejectStates, acceptStates and route as a member variables. We are using these variables when creating a Turing object. Let me mention about the functions included in Turing Class.

First we have isInRejectState, isInAcceptState functions in order to find out if we are in reject state or accept state.

We also have isAtLeftmostPosition, isAtRightmostPosition moveHeadLeft and moveHeadRight in order to make operations on current head. Also we have writeSymbol method that writes the given symbol to the the tape.

Now here is the most important function in Turing class. It is run function. What run function do is explained below:

* First we make a route list in order the see the path. We will print this list while printing the route to console.
* This function has boolean variable that checks if the Turing machine in a loop or not.
* We have a loop that continues when current state is not accept,loop or reject state. In this loop we change the state in each iteration and we write symbol also we change our direction to left or right.
* Also we check previousstate and current state if they are equal we set isStuck to true. Thats how we understand if there is a loop or not.

**Transition Class**

In this class we simply make transition function by using writerSymbol,moveDirection and nextState as member variables. We use those member variables while creating an object.

Text

Description automatically generated

**Direction Enum**

It is simply an enum that shows the directions left and right.

**Main Class**

In main class the input format a little bit different in this case.

* First, we decide the tape as a char array.
* After that we specify the mappings while writing inputs and states.
* We also define the accept and reject states.
* After deciding and specifying inputs we are ready to make a Turing object.
* While creating a Turing object, we run run function. Thanks to the run function we simulate a Turing machine with given inputs.
* For Printing the ROUT and RESULT i used isInAcceptState,isInRejectState functions and if its accepted or not i printed the route and result,in other conditions i also printed the route and result.

**Class Diagram**

Graphical user interface

Description automatically generated

**Conclusion**

In this simulation we have 3 choices either it will be accepted or rejected or it will be stuck in loop. We can see the changes in the resul tor rout if we change the inputs. So the program works correctly.