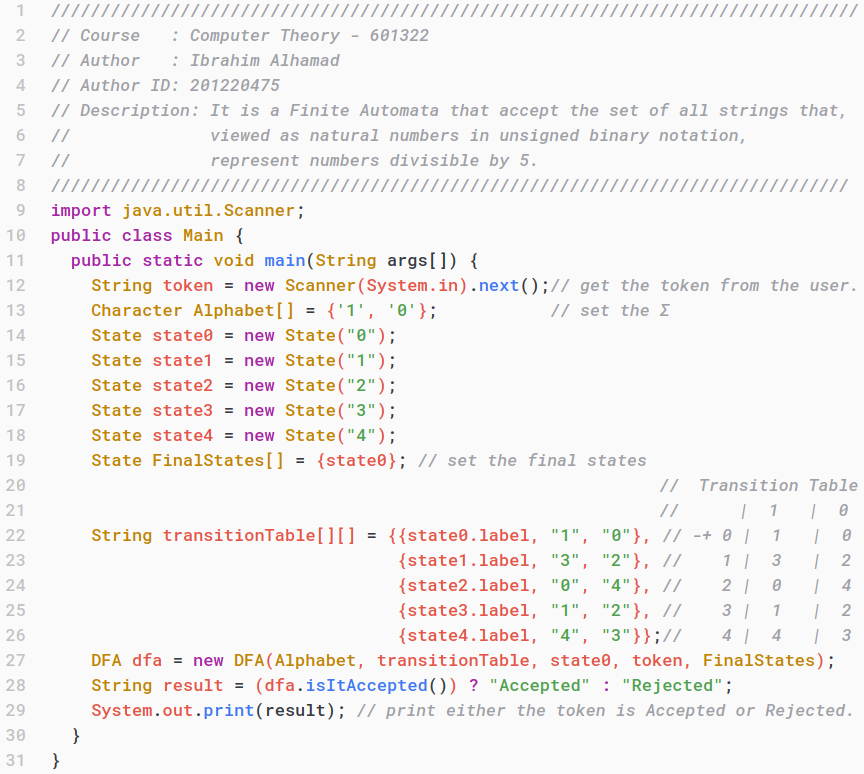
Assignment1

Computer Science

Faculty of Information Technology

**The Main Class :**



**The DFA Class:**

import java.util.Arrays;

import java.util.List;

public class DFA {

String transitionTable[][]; // transition table as a 2D array

List<Character> Alphabet;

State FinalStates[];

String currentState;

String token;

DFA(Character Alphabet[], String transitionTable[][],

State startState, String token, State FinalStates[]){

this.Alphabet = Arrays.asList(Alphabet); // convert the Σ into a List

this.transitionTable = transitionTable; // set the transition table

this.currentState = startState.label; // set the state label

this.FinalStates = FinalStates; // set the final states array

this.token = token;

}

boolean isItAccepted() {

if (Alphabet.contains(token.charAt(0))) { // check if the 1st Character

// is in the Alphabet

for(int index = 0;index < token.length();index++) {

int csAsIndex = Integer.parseInt(currentState);// copy the currentState

// as a Index

// if the current character is '1'

if (token.charAt(index) == Alphabet.get(0)) {

// check if the next state is not = current state

// if true set the current state = next state

if (!(transitionTable[csAsIndex][1] == currentState)) {

currentState = transitionTable[csAsIndex][1];

}

// if the current character is '0'

} else if (token.charAt(index) == Alphabet.get(1)) {

// check if the next state is not = current state

// if true set the current state = next state

if (!(transitionTable[csAsIndex][2] == currentState)) {

currentState = transitionTable[csAsIndex][2];

}

}

}

for (int i = 0; i < FinalStates.length; i++) {

//check if the current state is one of the final states

if (currentState.equals(FinalStates[i].label)){

return true;

}

}

}

return false;

}

}