JUNE 2022

FROM: **LE PHUONG THU**

ID: **BI11-258**

TO: LECTURER **GIANG ANH TUAN**

DEPARTMENT: **ICT**
(B2 / 2021-2022)

COMPUTATIONAL THEORY
FINAL EXAMINATION

THE QUESTION IS TO WRITE AN ESSAY TO ANSWER THE QUESTION:

Survey on the internet to see is it possible to have a Turing machine that can compute an ODE equation (ordinary differential equation). If yes, using your preferred programing language to code such a machine, then include your code in your essay with some example results.

TURING MACHINE:

The finite-state automata studied earlier in this chapter cannot be used as general models of computation.

Finite-state automata are able to recognize regular sets, but are not able to recognize many easy-to-describe sets, including {0n1n|n ≥ 0}, which computers recognize using memory.

Basically, a Turing machine consists of a control unit, which at any step is in one of the finitely many different states, together with a tape divided into cells, which is infinite in both directions.

Turing machines have read and write capabilities on the tape as the control unit moves back and forth along this tape, changing states depending on the tape symbol read.

RESEARCH RESULTS:

My answer is YES.

There are many types of Turing Machines. For example, Turing's very first example (Alan Turing 1937), 3-state Busy Beaver (Peterson 1988) ...

A Turing machine T = (S, I, f,s0) consists of a finite set S of states, an alphabet I containing the blank symbol B, a partial function f from S × I to S × I × {R, L}, and a starting state s0.

eg: *What is the final tape when the Turing machine T defined by the seven five-tuples (s0, 0,s0, 0, R),(s0, 1,s1, 1, R),(s0, B,s3, B, R),(s1, 0,s0, 0, R),(s1, 1,s2, 0, L),(s1, B,s3, B, R), and (s2, 1,s3, 0, R) is run on the tape shown in Figure (a) below?*

Let V be a subset of an alphabet I. A Turing machine T = (S, I, f, s0) recognizes a string x in V∗ if and only if T, starting in the initial position when x is written on the tape, halts in a final state. T is said to recognize a subset A of V∗ if x is recognized by T if and only if x belongs to A.

eg: *Find a Turing machine that recognizes the set of bit strings that have a 1 as their second bit, that is, the regular set (0 ∪ 1)1(0 ∪ 1)∗.*

Find a Turing machine that recognizes the set {0n1n|n ≥ 1}.

A decision problem asks whether statements from a particular class of statements are true. Decision problems are also known as yes-or-no problems.

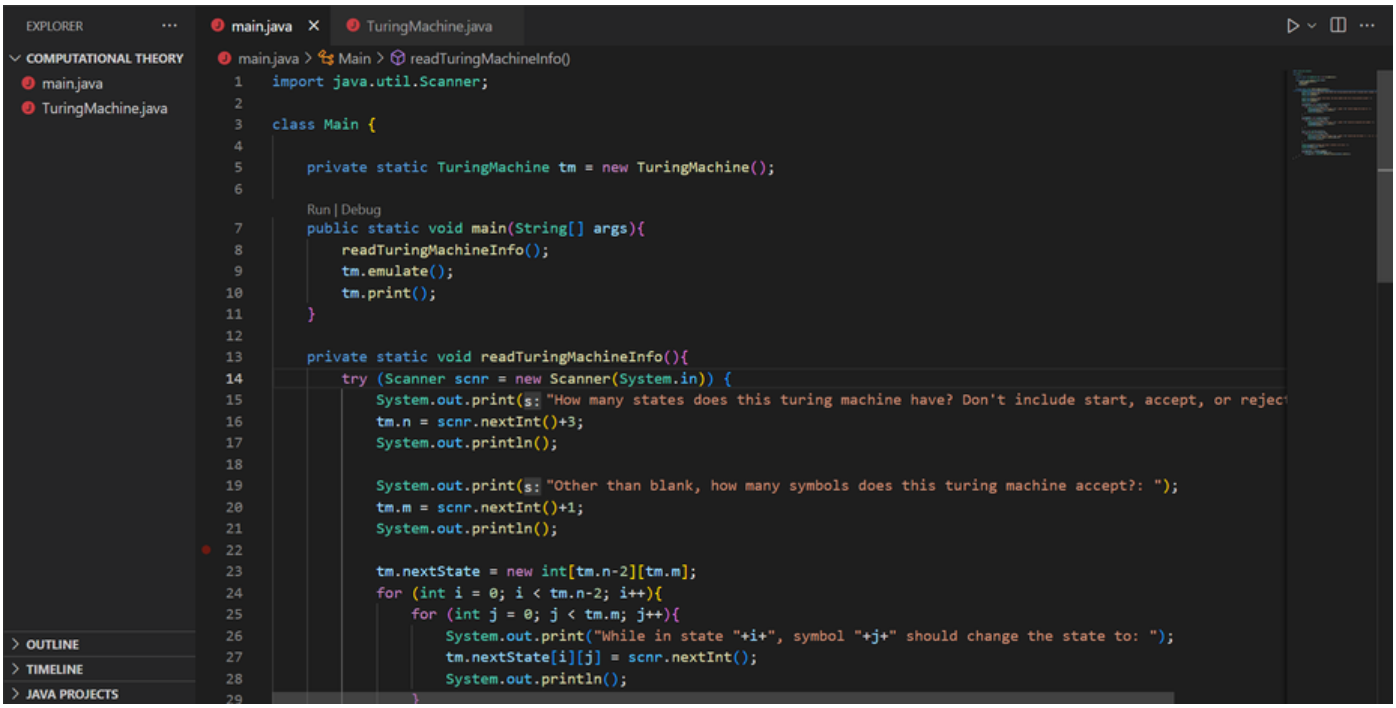
The halting problem is the decision problem that asks whether a Turing machine T eventually halts when given an input string x.

The halting problem is an unsolvable decision problem. That is, no Turing machine exists that, when given an encoding of a Turing machine T and its input string x as input, can determine whether T eventually halts when started with x written on its tape.

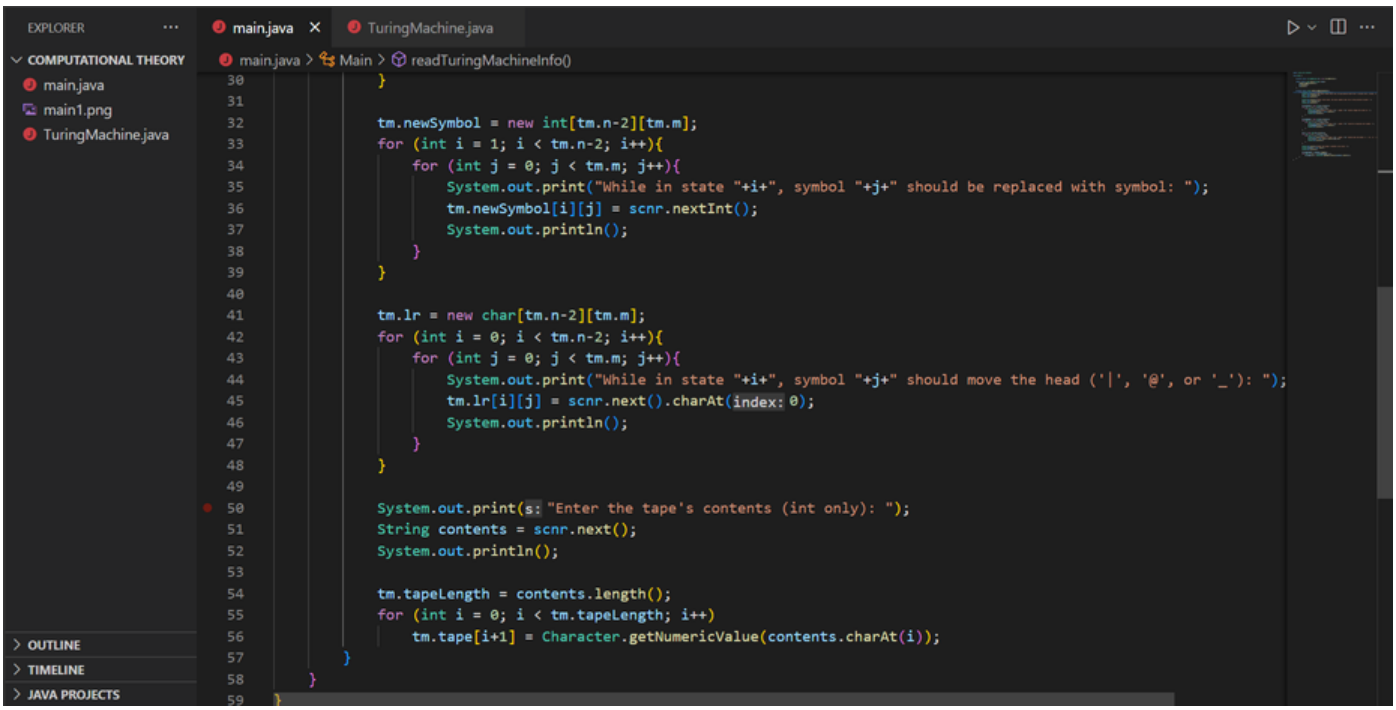
A decision problem is in P, the class of polynomial-time problems if it can be solved by a deterministic Turing machine in polynomial time in terms of the size of its input. That is, a decision problem is in P if there is a deterministic Turing machine T that solves the decision problem and a polynomial p(n) such that for all integers n, T halts in a final state after no more than p(n) transitions whenever the input to T is a string of length n. A decision problem is in NP, the class of non-deterministic polynomial-time problems if it can be solved by a non-deterministic Turing machine in polynomial time in terms of the size of its input. That is, a decision problem is in NP if there is a non-deterministic Turing machine T that solves the problem and a polynomial p(n) such that for all integers n, T halts for every choice of transitions after no more than p(n) transitions whenever the input to T is a string of length n.

CODE:

main.java

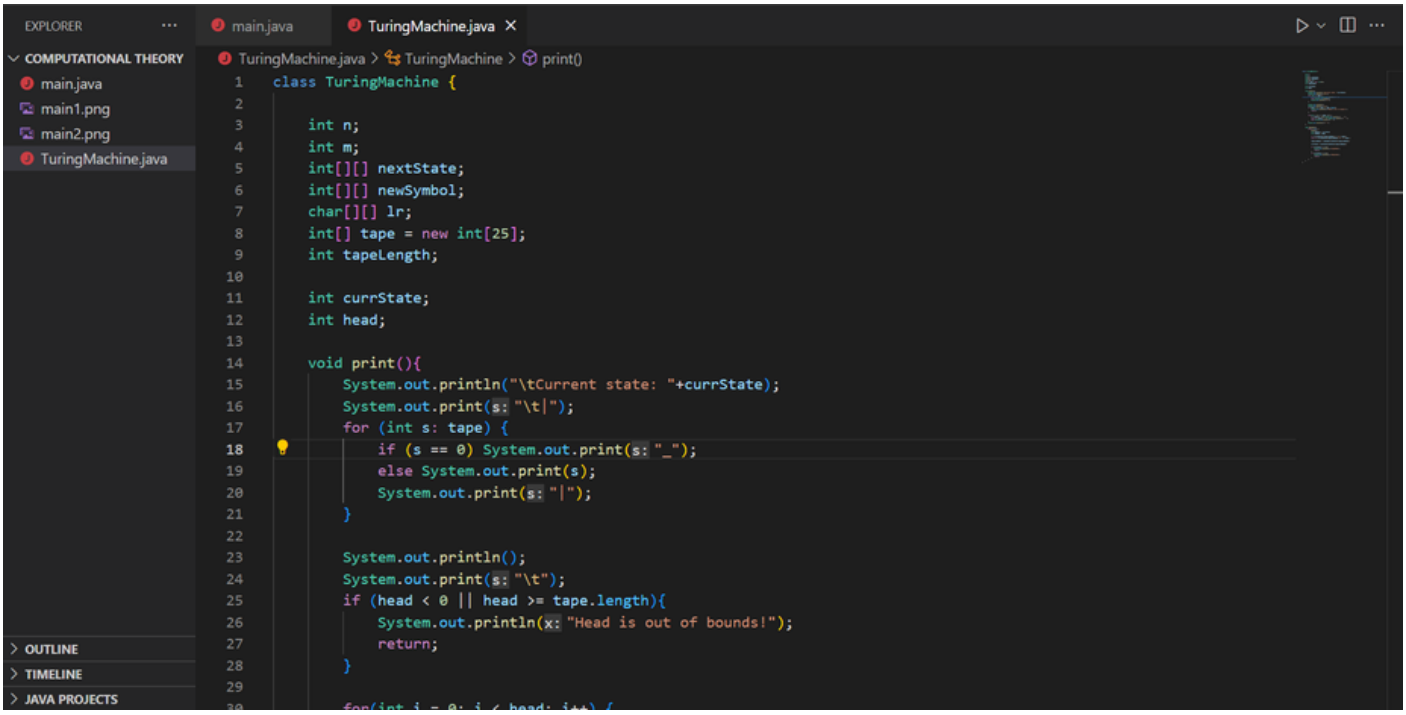


```java
import java.util.Scanner;

class Main {

    private static TuringMachine tm = new TuringMachine();

    Run | Debug
    public static void main(String[] args){
        readTuringMachineInfo();
        tm.emulate();
        tm.print();
    }

    private static void readTuringMachineInfo(){
        try (Scanner scnr = new Scanner(System.in)) {
            System.out.print(s: "How many states does this turing machine have? Don't include start, accept, or rejec
            tm.n = scnr.nextInt()+3;
            System.out.println();

            System.out.print(s: "Other than blank, how many symbols does this turing machine accept?: ");
            tm.m = scnr.nextInt()+1;
            System.out.println();

            tm.nextState = new int[tm.n-2][tm.m];
            for (int i = 0; i < tm.n-2; i++){
                for (int j = 0; j < tm.m; j++){
                    System.out.print("While in state "+i+", symbol "+j+" should change the state to: ");
                    tm.nextState[i][j] = scnr.nextInt();
                    System.out.println();
```

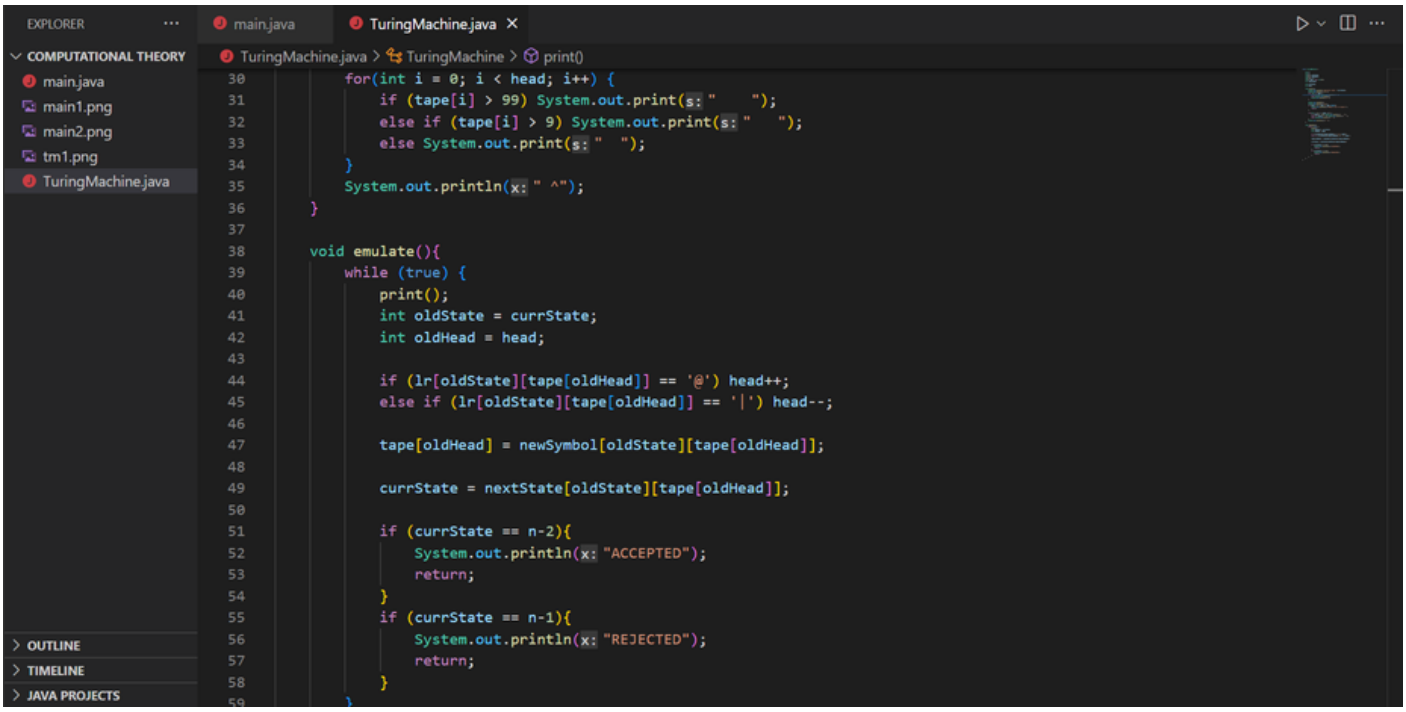


```java
            }

            tm.newSymbol = new int[tm.n-2][tm.m];
            for (int i = 1; i < tm.n-2; i++){
                for (int j = 0; j < tm.m; j++){
                    System.out.print("While in state "+i+", symbol "+j+" should be replaced with symbol: ");
                    tm.newSymbol[i][j] = scnr.nextInt();
                    System.out.println();
                }
            }

            tm.lr = new char[tm.n-2][tm.m];
            for (int i = 0; i < tm.n-2; i++){
                for (int j = 0; j < tm.m; j++){
                    System.out.print("While in state "+i+", symbol "+j+" should move the head ('|', '@', or '_'): ");
                    tm.lr[i][j] = scnr.next().charAt(index: 0);
                    System.out.println();
                }
            }

            System.out.print(s: "Enter the tape's contents (int only): ");
            String contents = scnr.next();
            System.out.println();

            tm.tapeLength = contents.length();
            for (int i = 0; i < tm.tapeLength; i++)
                tm.tape[i+1] = Character.getNumericValue(contents.charAt(i));
        }
    }
}
```

TuringMachine.java



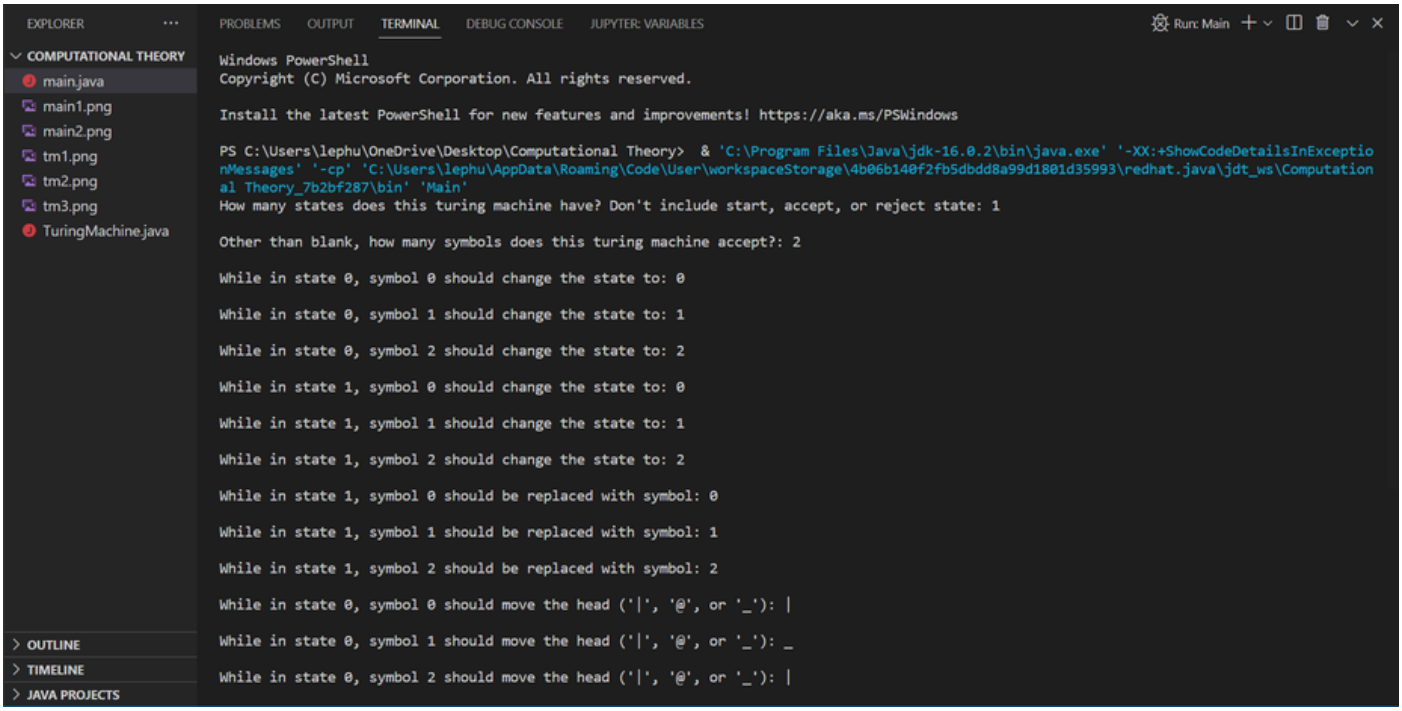
```
class TuringMachine {

    int n;
    int m;
    int[][] nextState;
    int[][] newSymbol;
    char[][] lr;
    int[] tape = new int[25];
    int tapeLength;

    int currState;
    int head;

    void print(){
        System.out.println("\tCurrent state: "+currState);
        System.out.print(s: "\t|");
        for (int s: tape) {
            if (s == 0) System.out.print(s: "_");
            else System.out.print(s);
            System.out.print(s: "|");
        }

        System.out.println();
        System.out.print(s: "\t");
        if (head < 0 || head >= tape.length){
            System.out.println(x: "Head is out of bounds!");
            return;
        }
```

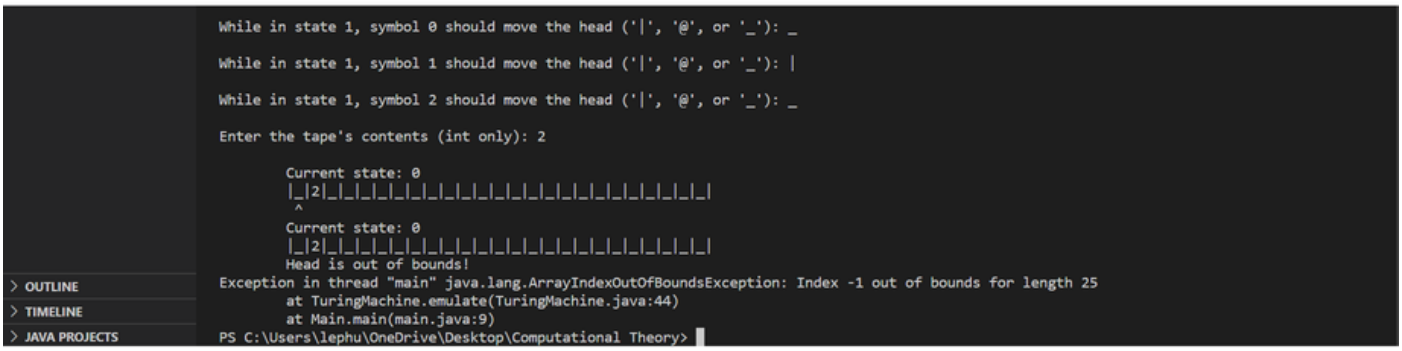


```
        for(int i = 0; i < head; i++) {
            if (tape[i] > 99) System.out.print(s: "    ");
            else if (tape[i] > 9) System.out.print(s: "   ");
            else System.out.print(s: "  ");
        }
        System.out.println(x: " ^");
    }

    void emulate(){
        while (true) {
            print();
            int oldState = currState;
            int oldHead = head;

            if (lr[oldState][tape[oldHead]] == '@') head++;
            else if (lr[oldState][tape[oldHead]] == '|') head--;

            tape[oldHead] = newSymbol[oldState][tape[oldHead]];

            currState = nextState[oldState][tape[oldHead]];

            if (currState == n-2){
                System.out.println(x: "ACCEPTED");
                return;
            }
            if (currState == n-1){
                System.out.println(x: "REJECTED");
                return;
            }
```

```
        }
    }
}
```

RESULT:





CONTACT INFORMATION:

- Email: thulp.bi11-258@st.usth.edu.vn
- ICT Department