Homework Assignment #3
Due: Monday, March 2, 2026 at 7:00 p.m.

Design a Turing machine that decides whether the input string is in $\{\mathtt{0}^n\mathtt{1}^{2n} : n \geq 0\}$.

For this question, you will submit your solution electronically as a text file that contains a description of the Turing machine in York University Turing Machine File Format (YUTMFF), which is described below.

YUTMFF

The Turing machines described in YUTMFF use the following conventions, as described in the lectures.

- They use a 1-way infinite tape.
- The tape alphabet has two different special symbols, $\triangleright$ and $\sqcup$ that are not part of the input alphabet.
- Initially, if the input string is w, the tape contains $\triangleright w$ at the left end of the tape, and the rest of the tape contains only $\sqcup$ symbols. The head of the Turing machine is initially positioned at the first character of the input string w (i.e., at the tape's second square).
- Whenever the Turing machine sees the $\triangleright$ symbol, it must leave it unchanged and move right (but it can change state).

We also make some naming conventions. We assume that the state set of the Turing machine is $Q = \{q_0, q_1, \ldots, q_{n-1}\}$ where $n \geq 3$ and the tape alphabet of the Turing machine is $\Gamma = \{c_0, c_1, \ldots, c_{m-1}\}$ where $m \geq 3$. We also assume that q_0 is the initial state, q_{n-2} is the accepting state and q_{n-1} is the rejecting state. We assume that the input alphabet is $\Sigma = \{c_0, c_1, \ldots, c_{k-1}\}$ where $0 \leq k \leq m-2$ and $c_{m-2} = \sqcup$ and $c_{m-1} = \triangleright$.

We now explain how to describe, using YUTMFF, a Turing machine that follows the conventions described above. An example appears below. The first line of the file contains the three integers n, m, and k, separated by single spaces. (Recall that these are the sizes of the state set, tape alphabet and input alphabet, respectively.)

Each character in the tape alphabet has a name. The second line of the file contains $m-2$ strings separated by single spaces that give the names of the characters $c_0, c_1, \ldots, c_{m-3}$. We use the name `blank` to represent $c_{m-2} = \sqcup$ and `leftend` to represent $c_{m-1} = \triangleright$.

The third line contains a non-negative integer T.

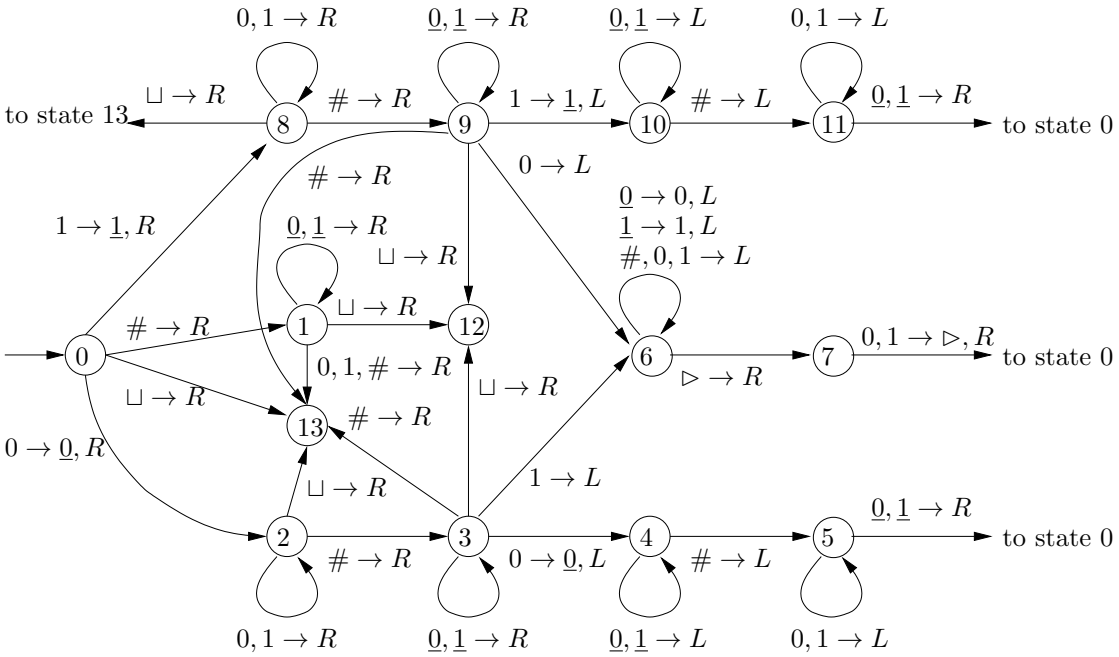
Following this, there are T lines. Each of these remaining lines of the description contains five items i, a, i', a', d separated by single spaces, where i and i' are integers with $0 \leq i \leq n-3$ and $0 \leq i' \leq n-1$ (inclusive), a and a' are names of characters in the tape alphabet and d is a single character that is either L or R. This line indicates that $\delta(q_i, a) = (q_{i'}, a', d)$. No two lines should have the same i and a. No transitions are given for situations when the machine is in state q_{n-2} or q_{n-1} since those are the accepting and rejecting states. If

no transition is given to describe $\delta(q_i, a)$ for a non-halting state q_i, then it is assumed that $\delta(q_i, a) = (q_i, a, R)$.

Some Java code will be posted on the course web page to assist you with this assignment. If you run the Java programme TM.java on your input file, it will test it on a number of input strings. If it gives you an error message, there is something wrong with your TM description file. Your solution will be tested automatically to assign you a grade, so if you submit a solution containing a syntax error, you may be assigned a grade of 0. Therefore, you really should **make sure that TM.java runs correctly on your file**.

Example of YUTMFF

Consider the following Turing machine, which decides the language $\{uvw\#v : u, v, w \in \{0, 1\}^*\}$.



It can be encoded as the following file in YUTMFF.

```
14 7 3
0 1 # 0_ 1_
42
0 0 2 0_ R
0 1 8 1_ R
0 # 1 # R
0 blank 13 blank R
1 0 13 0 R
1 1 13 1 R
1 # 13 # R
1 blank 12 blank R
2 # 3 # R
2 blank 13 blank R
```

```
3 0 4 0_ L
3 1 6 1 L
3 # 13 # R
3 blank 12 blank R
4 0_ 4 0_ L
4 1_ 4 1_ L
4 # 5 # L
5 0 5 0 L
5 1 5 1 L
5 0_ 0 0_ R
5 1_ 0 1_ R
6 0_ 6 0 L
6 1_ 6 1 L
6 # 6 # L
6 0 6 0 L
6 1 6 1 L
6 leftend 7 leftend R
7 0 0 leftend R
7 1 0 leftend R
8 blank 13 blank R
8 # 9 # R
9 0 6 0 L
9 1 10 1_ L
9 # 13 # R
9 blank 12 blank R
10 0_ 10 0_ L
10 1_ 10 1_ L
10 # 11 # L
11 0 11 0 L
11 1 11 1 L
11 0_ 0 0_ R
11 1_ 0 1_ R
```

Submission instructions

Type your solution in a **plain text file** named a3.txt. (See assignment 1 for instructions on plain text files. Do not submit a pdf or a word document!) To submit your file, run the following command on an EECS machine:

`submit 2001M a3 a3.txt`

If you wish to declare that you have discussed your solution with other students, type your declaration in a plain text file called declaration.txt and submit it using the command:

`submit 2001M a3 declaration.txt`

You can also use the web interface to submit your solution at

`https://webapp.eecs.yorku.ca/submit` .