

18CSCI03H

Theory of Computing

Assignment 2

By Ahmed Hossam ElSabbagh

147176

Turing Machine One

The given of the equation 7x+1 for every x>=0 can be described through the following Turing Machine diagram.



This machine consists of 12 states, including the entry state q0 and the accepting state q11 colored in blue.

The input of this machine consists of an infinite tape starting with the following sequence #X#. Where X is any number of x ones, for example if x=3, X=111, therefore the input is #111#.

The machine works following this logic:

1) The index of the tape points to 0, or #, it than ignores it and goes to starting state q0 and moves the index to 1. #111#

2) For the first 1 the index is pointing at, the machine reads it at state q0, writing x and moves right, than it goes state q1. #x11#

3) The machine in q1 reads all the ones and write them back, logically ignoring them. #x11#, #x11#.

4) When finally the index reaches the #, the machine moves to q2, reading and writing # to logically ignore it. (b refers to blank) (The code logically adds blanks for every time the index reaches the end of the tape) #x11#b.

5) q2 then reads and writes all the following ones to logically ignore it and move the index right. #x11#b.

6) When finally the index reaches a blank it reads it and writes 1, then it moves right. #x11#1b, going to q3.

7) From q3 to q7, the states has only one transition, read blanks, writes 1s, and move right, go to the following states q4 q5 q6 q7. #x11#11111b.

8) When reaching state q7, the machine reads the blank at the index and moves left, and goes to q8. #x11#111111b.

9) In q8, the index moves reads and write 1s to ignore it and moves left. #x11#111111b, #x11#111111b, #x11#111111b, #x11#111111b, #x11#111111b.

10) When reading, # in q8, it continues left, ignoring # and going to q9, in q9, it goes back and ignores 1s until it reaches x. #x11#111111b, #x11#111111b, and #x11#111111b. When x is reached, the machine moves to q0 and the index moves right. #x11#111111b.

11) The machine than repeats the cycle initiated in step 2. Also following step 6, this is an example of what happens. #xx1#111111b, #xx1#111111b, #xx1#111111b, #xx1#111111b, #xx1#111111b, #xx1#111111b, #xx1#111111b.

12) This cycle is repeated until all ones behind the second # is turned into x. It should be noted that the total number of 1s should be 6\*x. In this case there are 18 ones. #xxx#111111111111111111b.

13) Going back to step 12 the machine should now be in q0, with the index pointing to the second #, therefore it should go left, turning the # into 1 and going to q10 #xxx1111111111111111111b.

14) When in q10, the machine loops on the previous x and turn them into 1s, #1111111111111111111111b. It than goes to q11 and turns the last # into 0. 01111111111111111111111b. It can be noted that the total number of 1s go as follow, 3\*6 from the first steps, which is 18, +1 that turned from the second # in step 13, a total of 19, and +3 from the x that turned into 1s in step 14, for total of 22, which is 7\*3 +1 or 21+1 or 22.

There are two variations of this machine that were created, the code of which can be found in the file.

This version puts 7 1s then turns x into 0s like this 00001111111111111111111111b (13 states).



While this one turns x into 1 when it comes back, for example at the beginning of the second loop, this should be the output #1x1#111111b, which is less computing time but leaves the # at the start. The final output should look like this #1111111111111111111111b (11 states)

