Name: Cem Meriç Şefikoğulları

ID: 2448850

Q1)

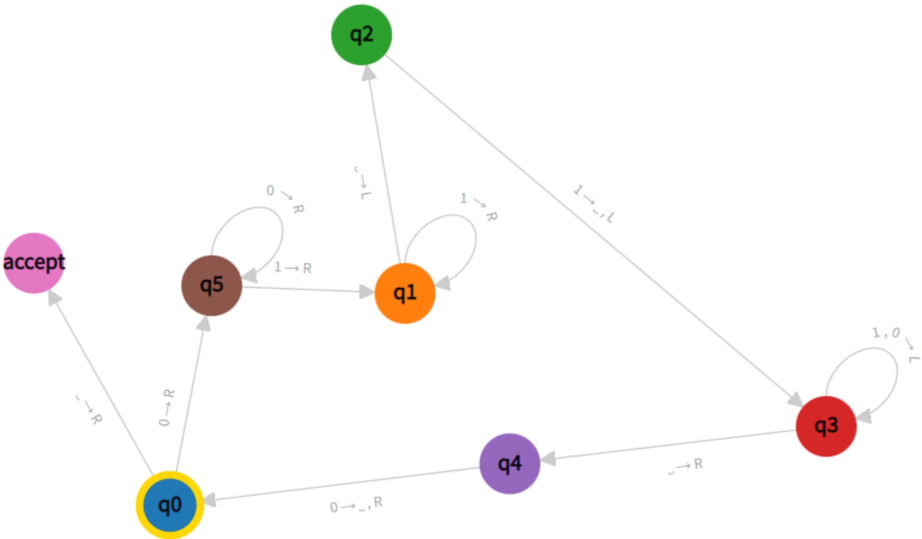
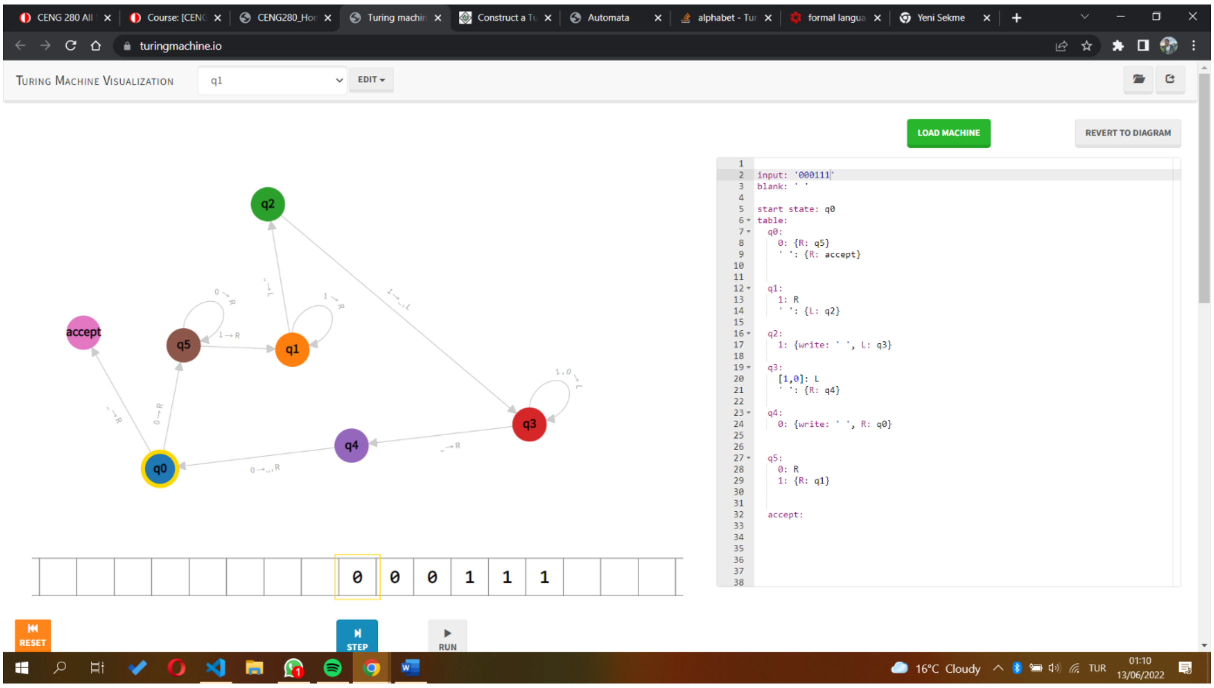
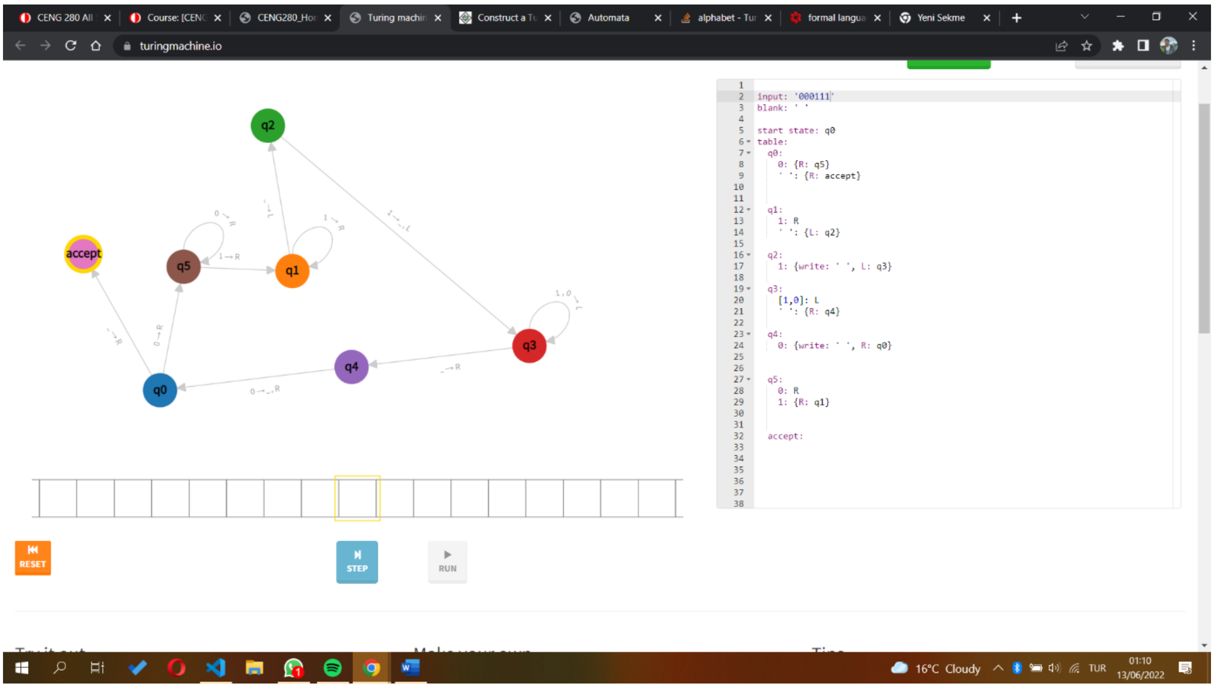


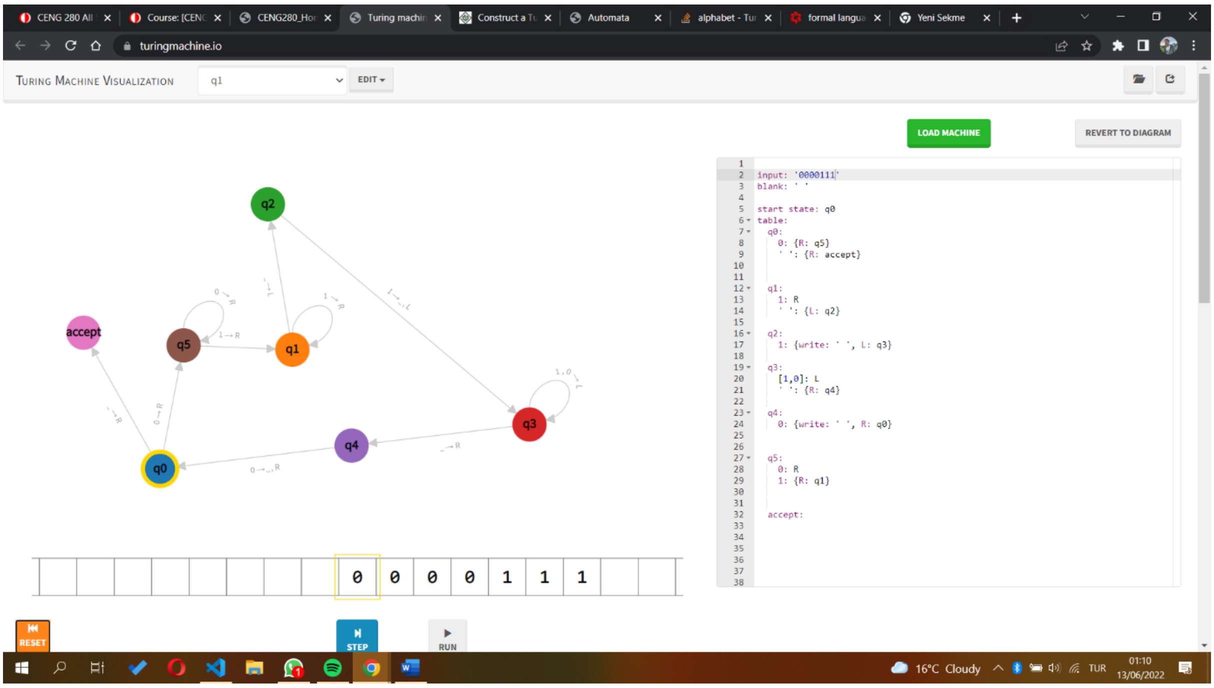
Figure of machine



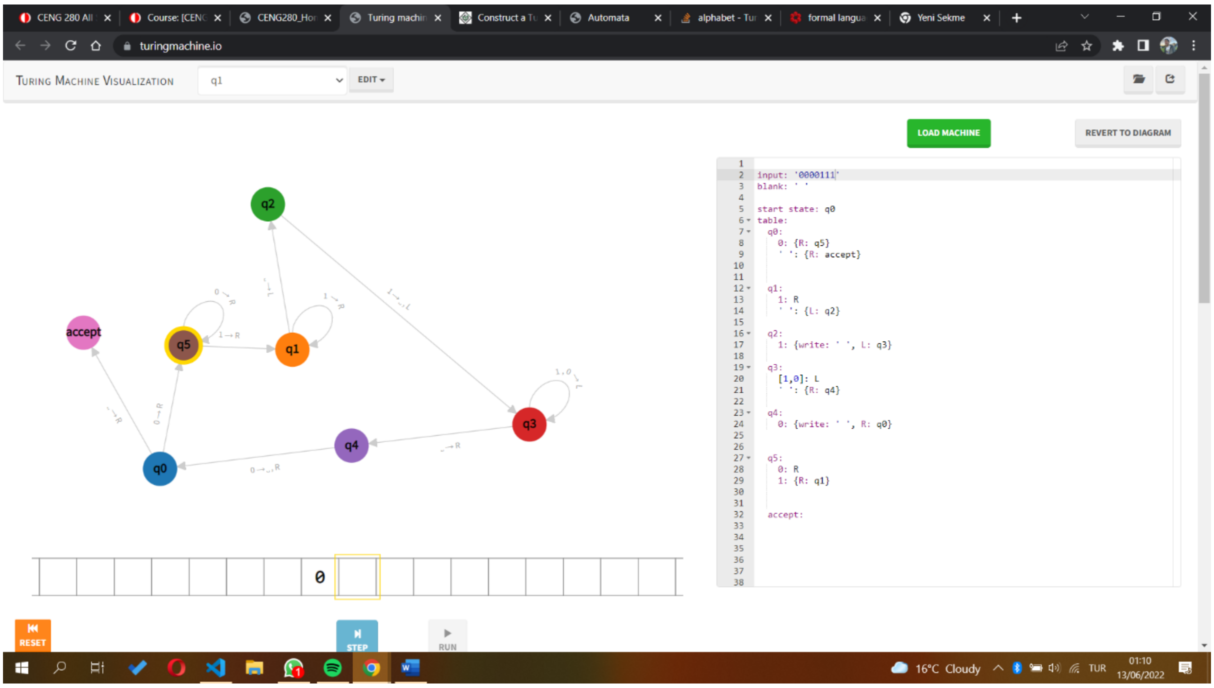
Initial state: 000111,q0



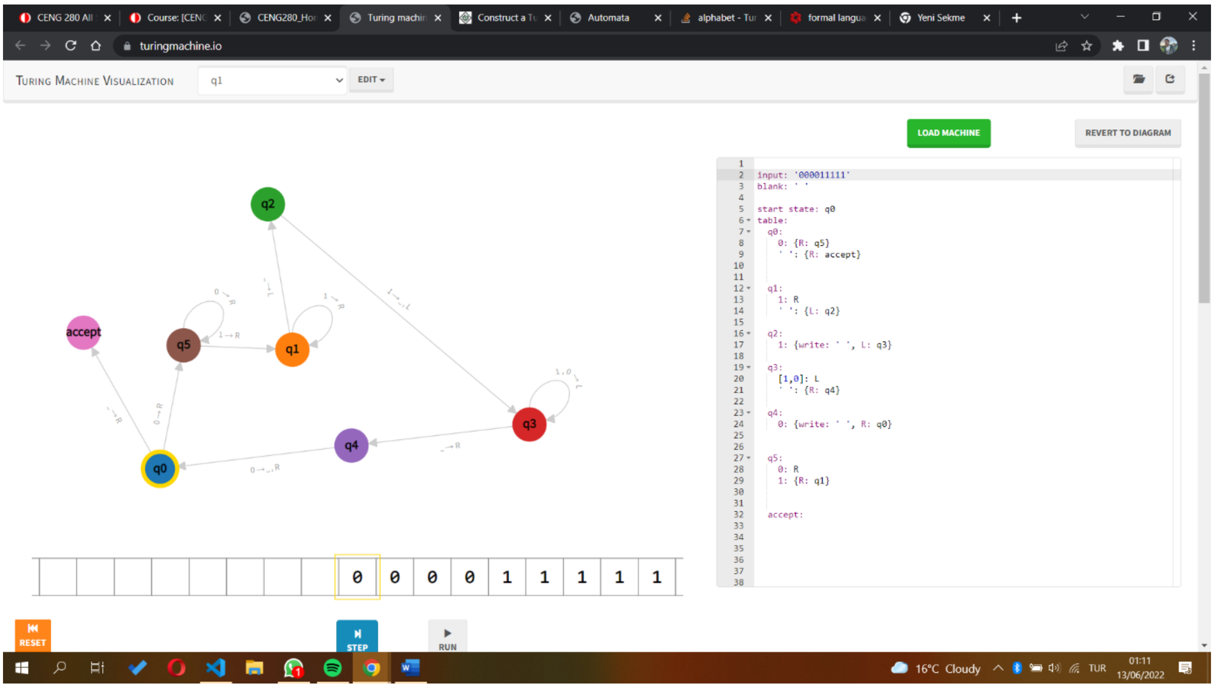
End state: _,accept state accepted



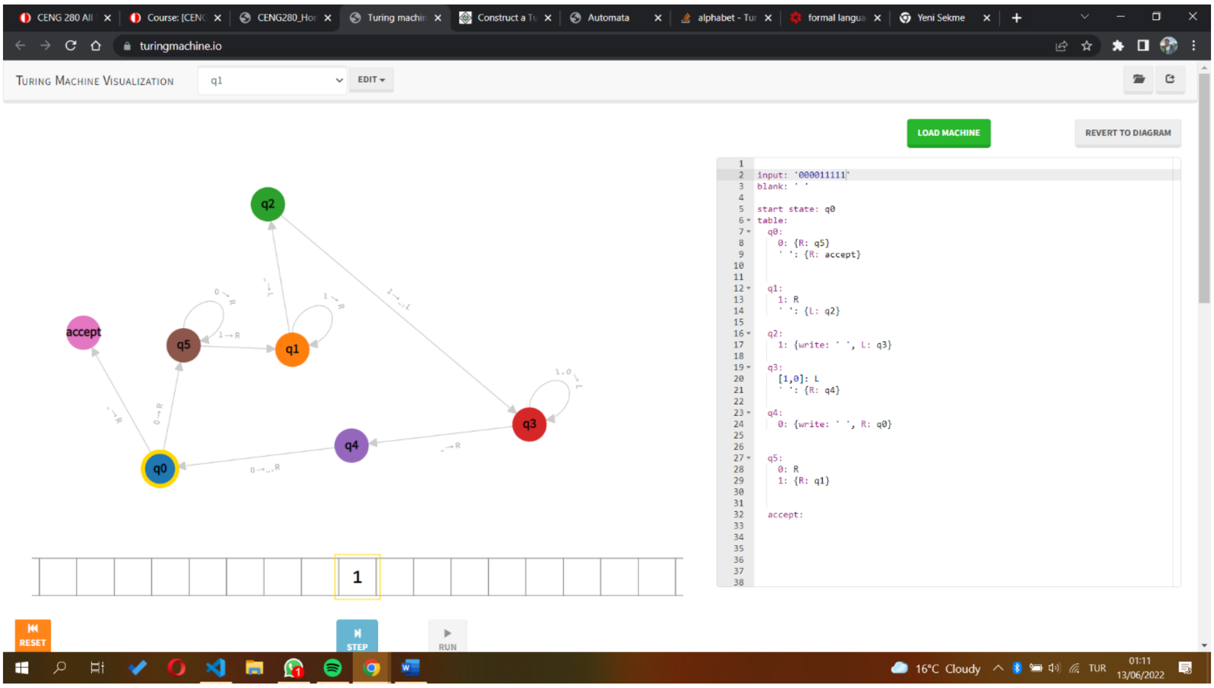
Initial state:0000111,q0



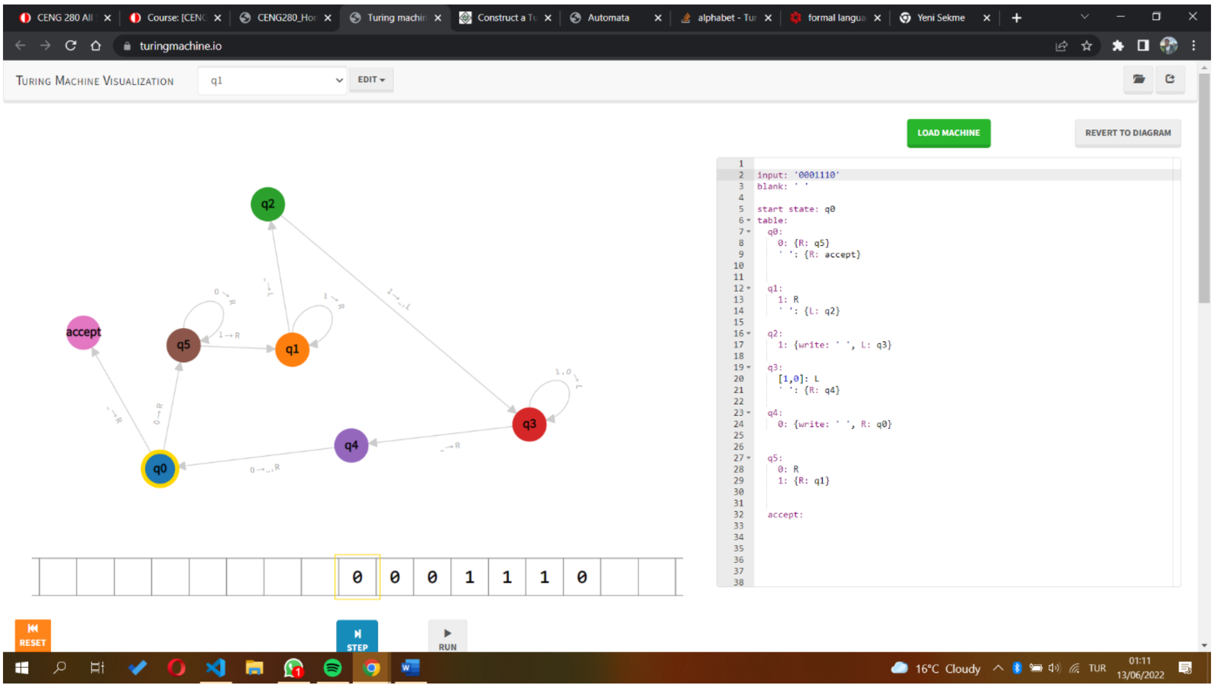
End state: 0,q5 rejected



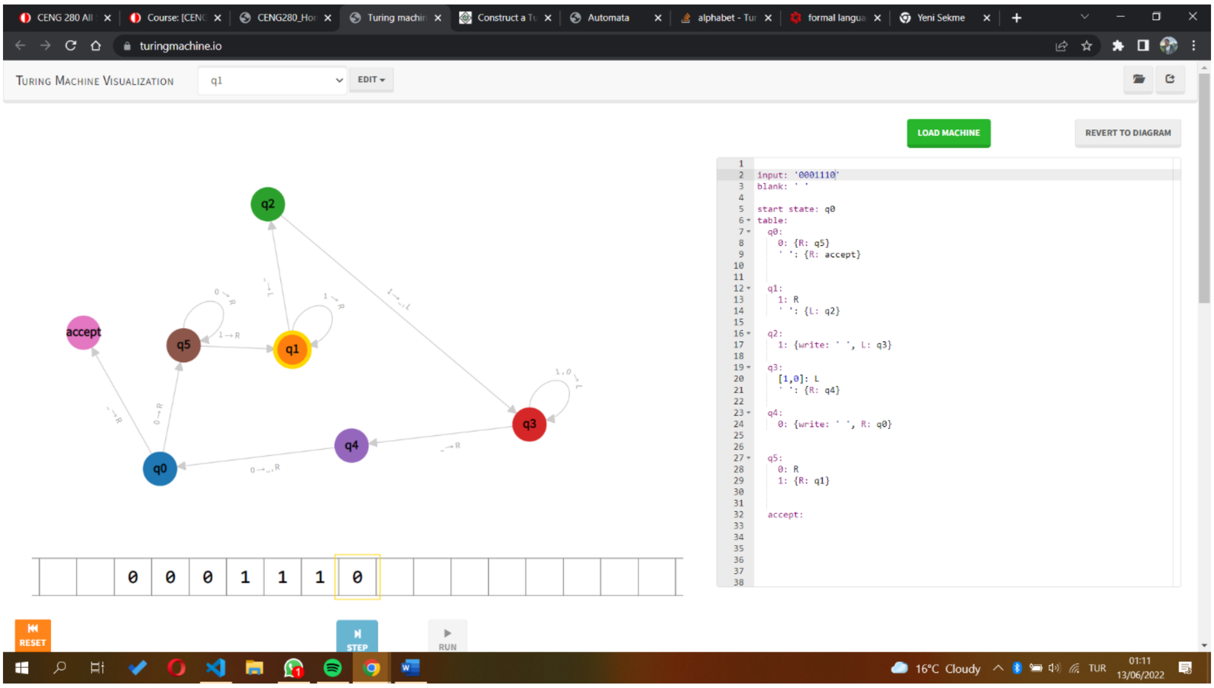
Initial state: 000011111,q0



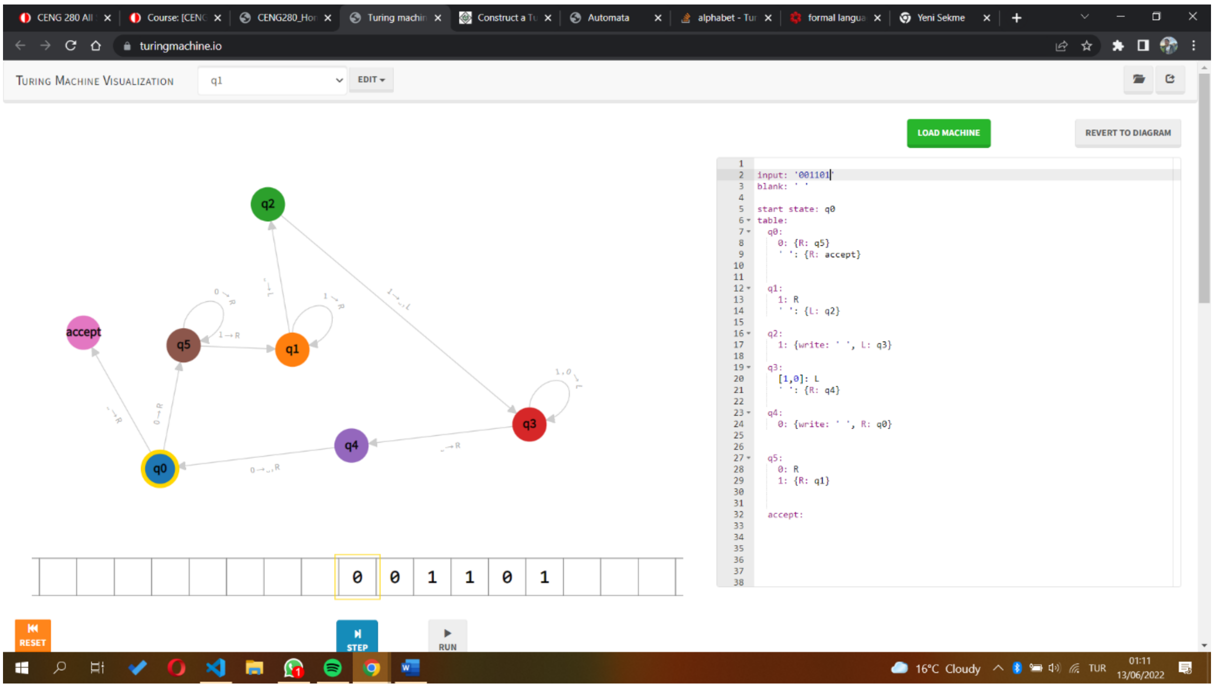
End state: 1,q0 rejected



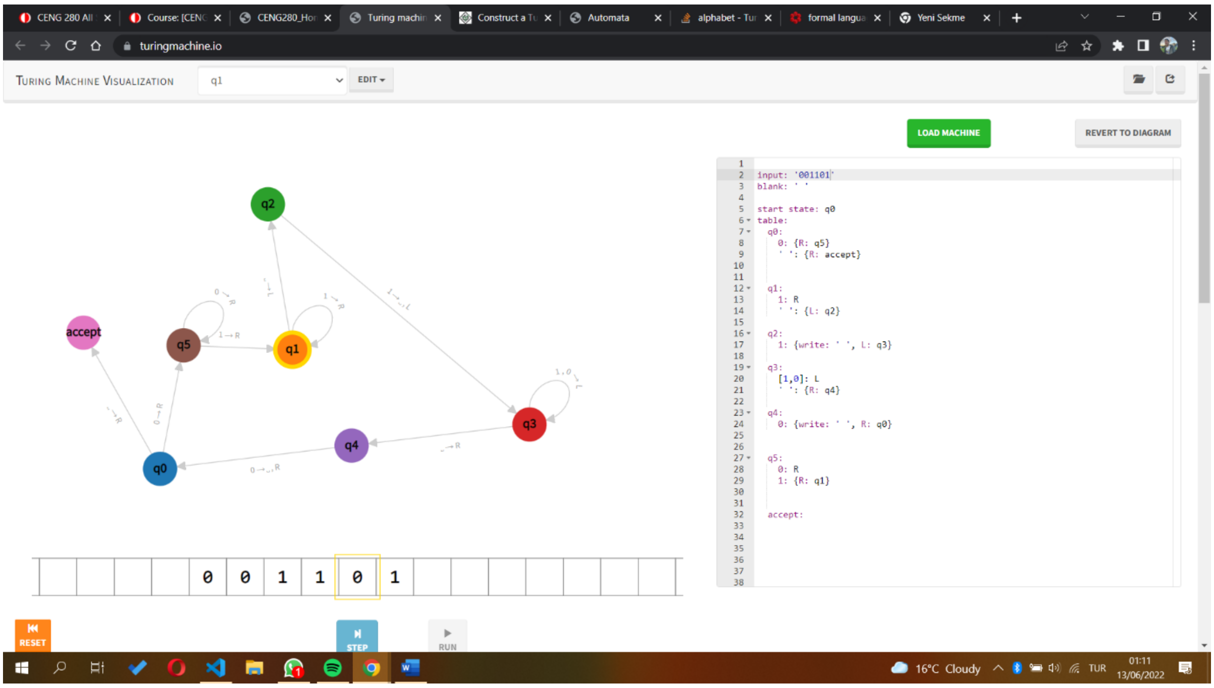
Initial state: 0001110 , q0



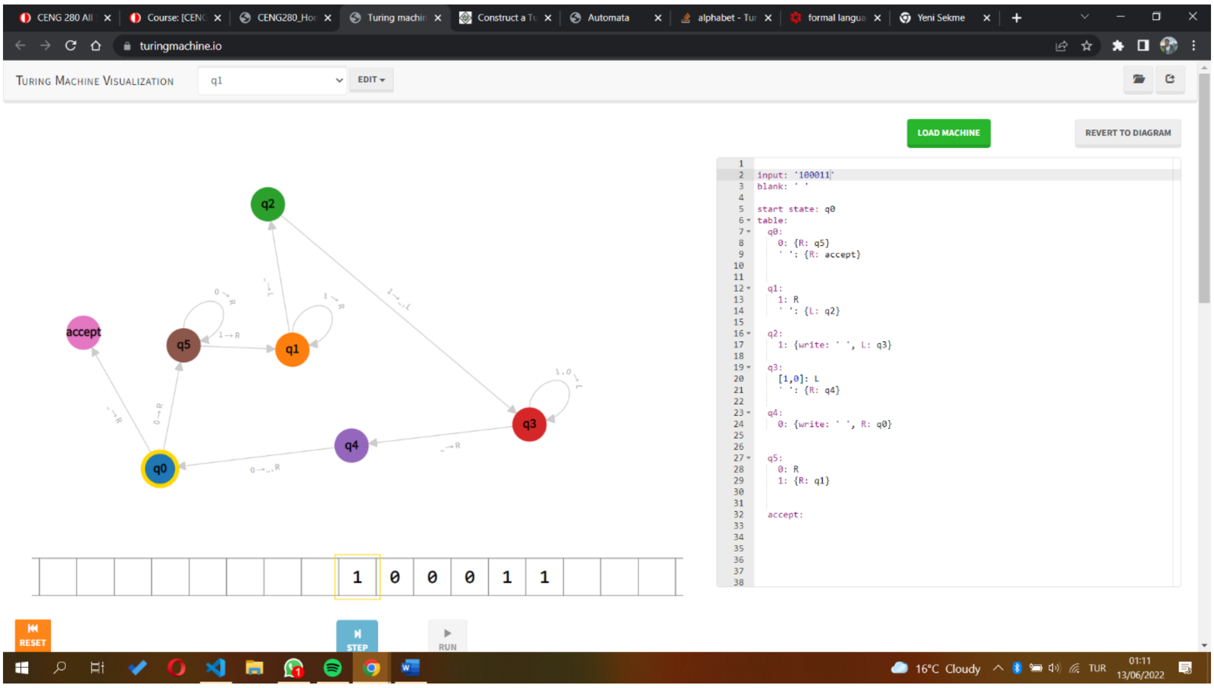
End state: 0001110, q1 rejected



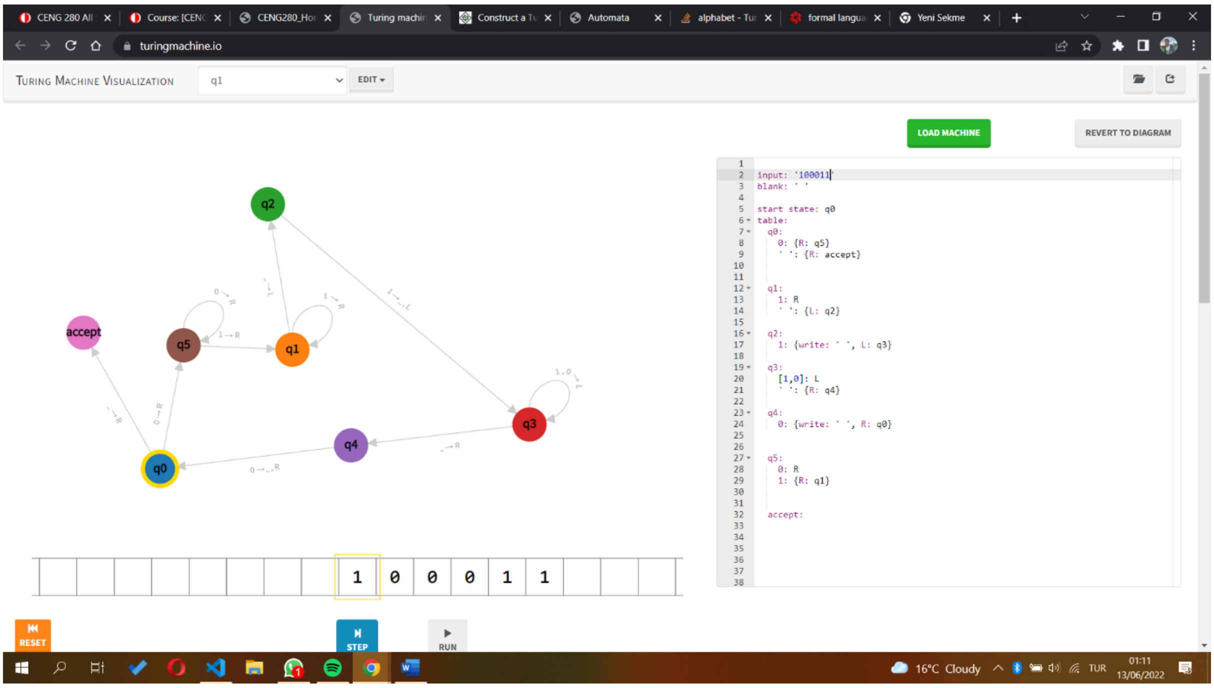
Initial state:001101, q0



End state: 001101,q1 rejected



Initial state: 100011, q0



End state: 100011, q0 rejected

Description of the states:

Start State q0:

If symbol 0 move right go to the q5 state

If symbol blank move right go to accept state

State q1:

If symbol q1 move right stay in the q1 state

If symbol blank move left go to the q2 state

State q2:

If symbol 1 replace it by blank move left go to the q3 state

State q3:

If symbol 1 or 0 move left stay in the q3 state

If symbol blank move right go to the q4 state

State q4:

If symbol 0 replace it by blank move right go to the q0 state

State q5:

If symbol 0 move right stay in the q5 state

If symbol 1 move right go to the state q1

Accept state:

Accept the string

Q2)

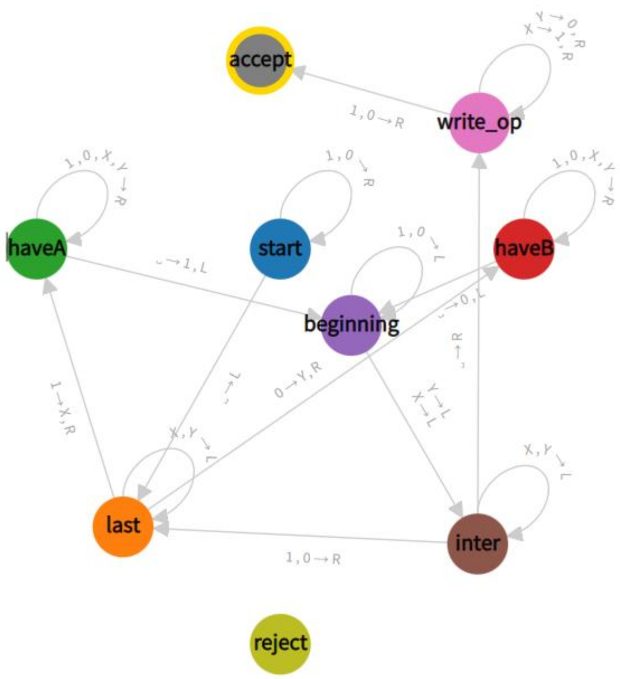
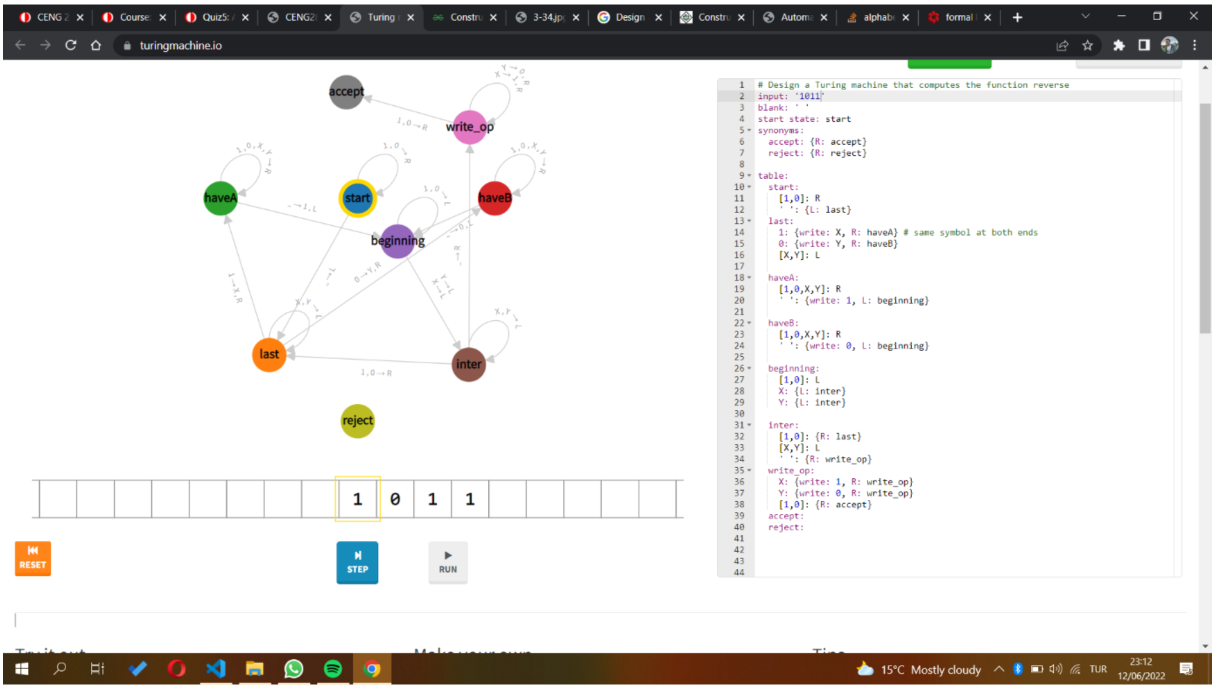
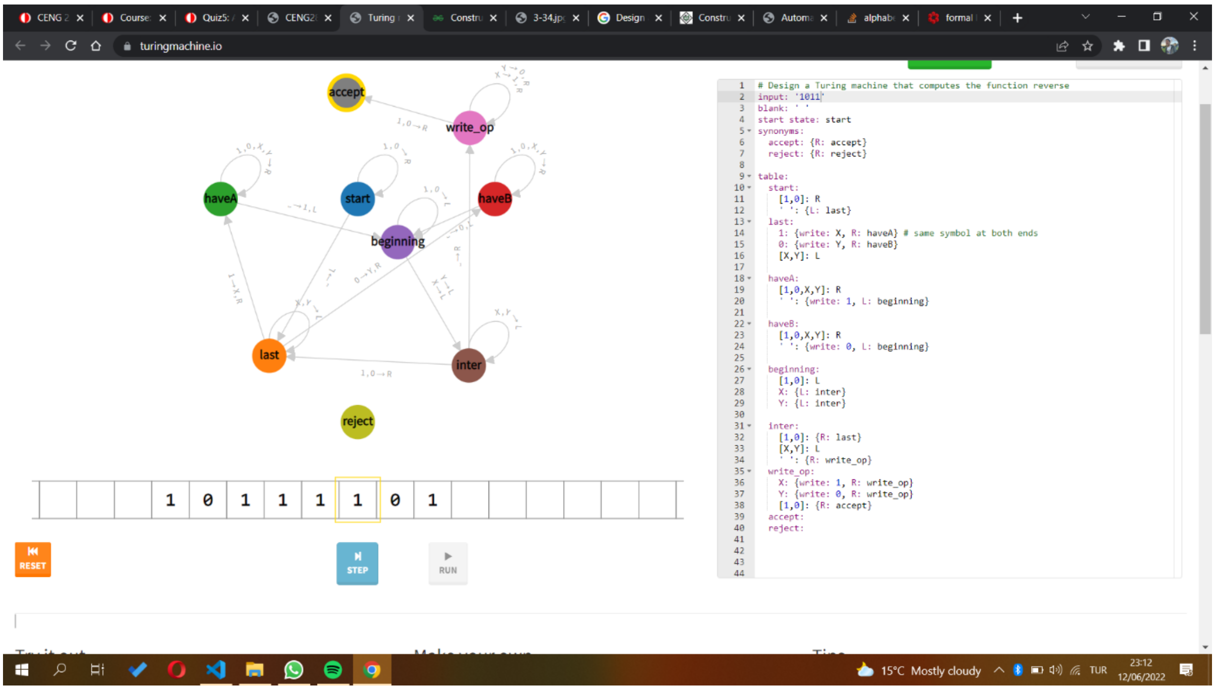


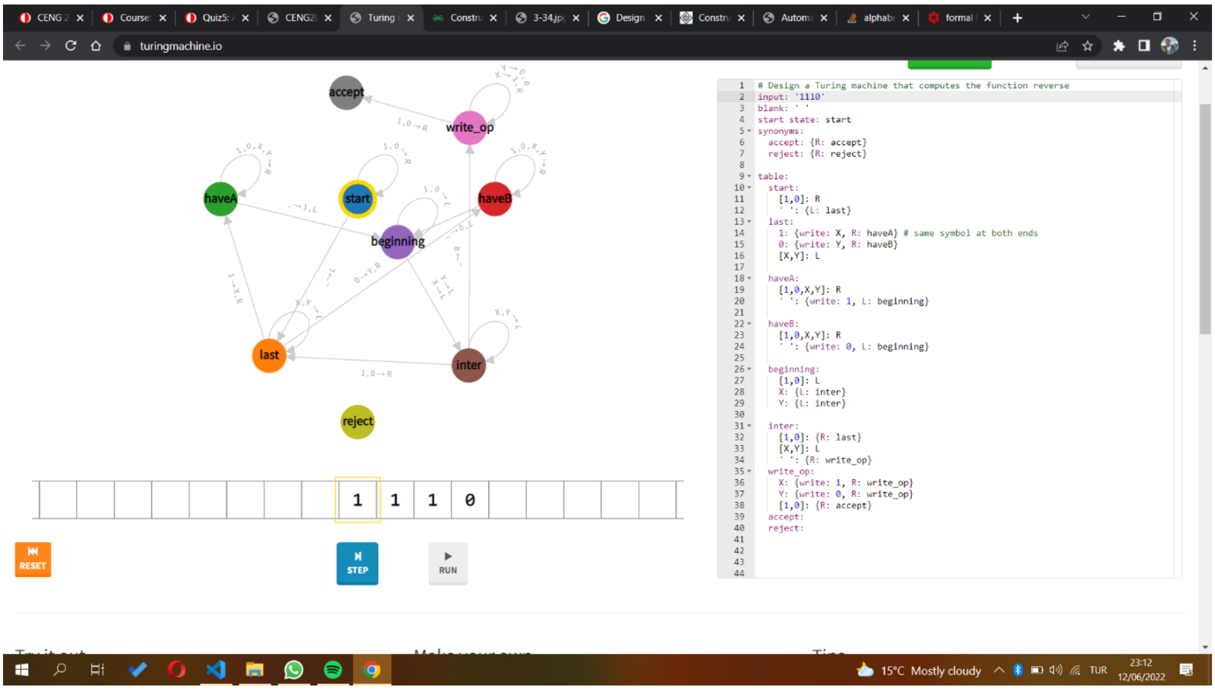
Figure of Machine



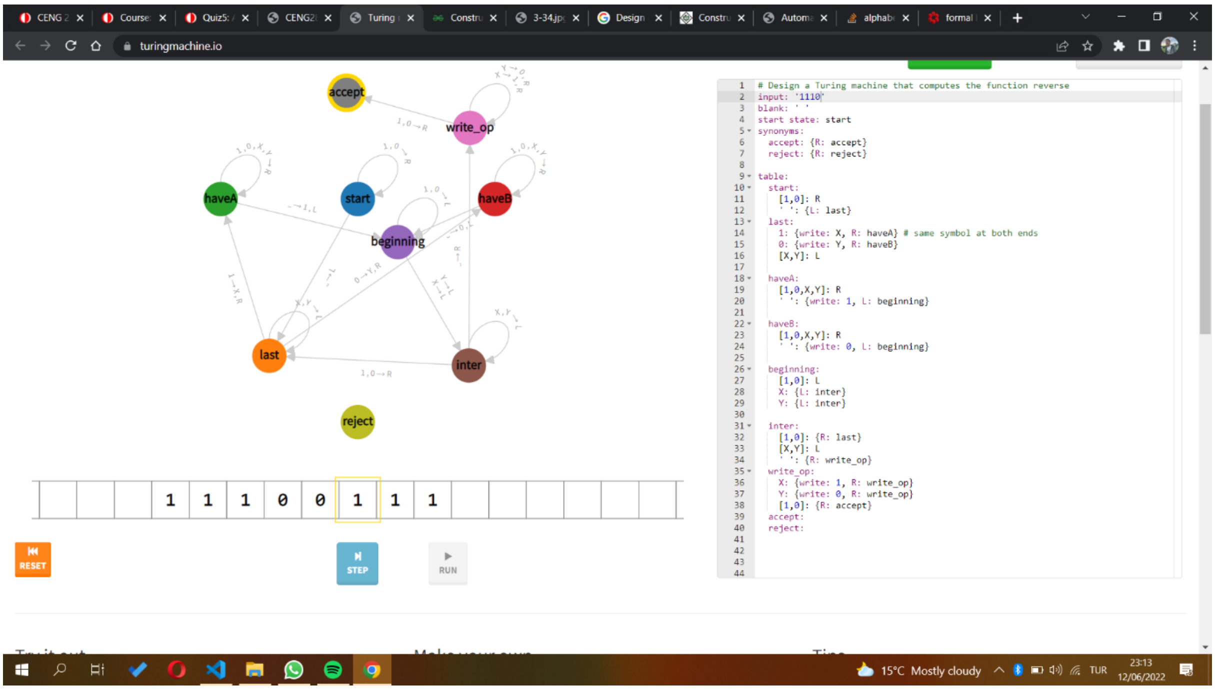
Initial state 1011



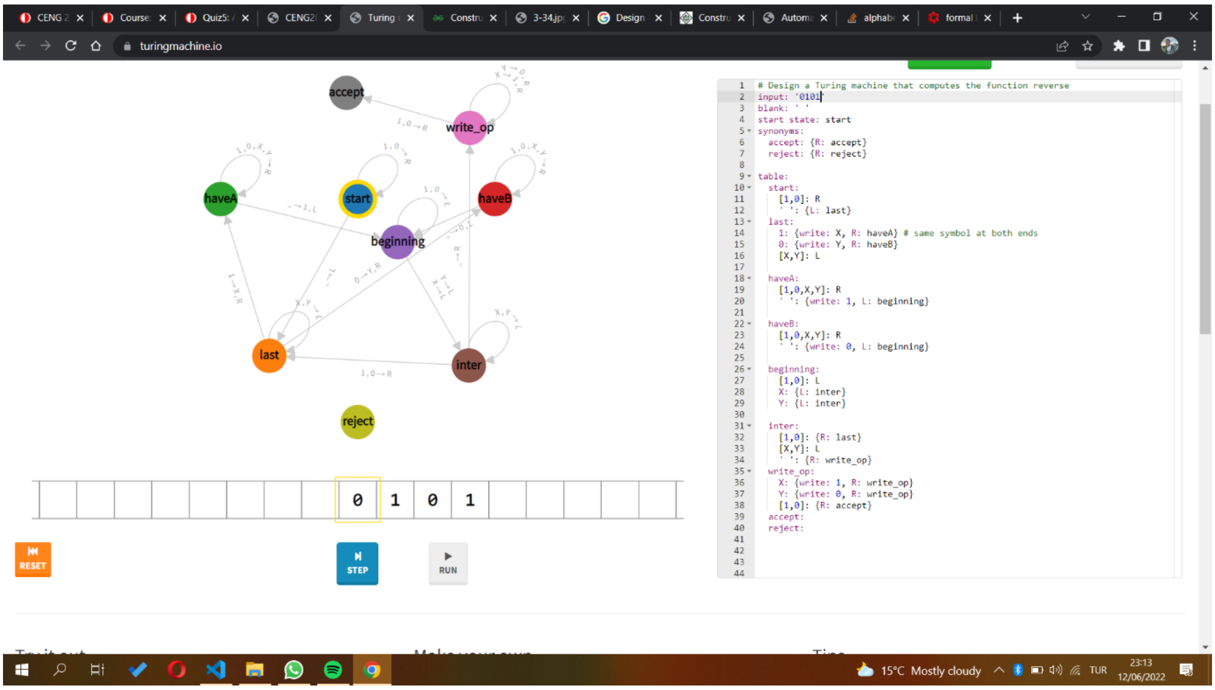
End state: 10111101



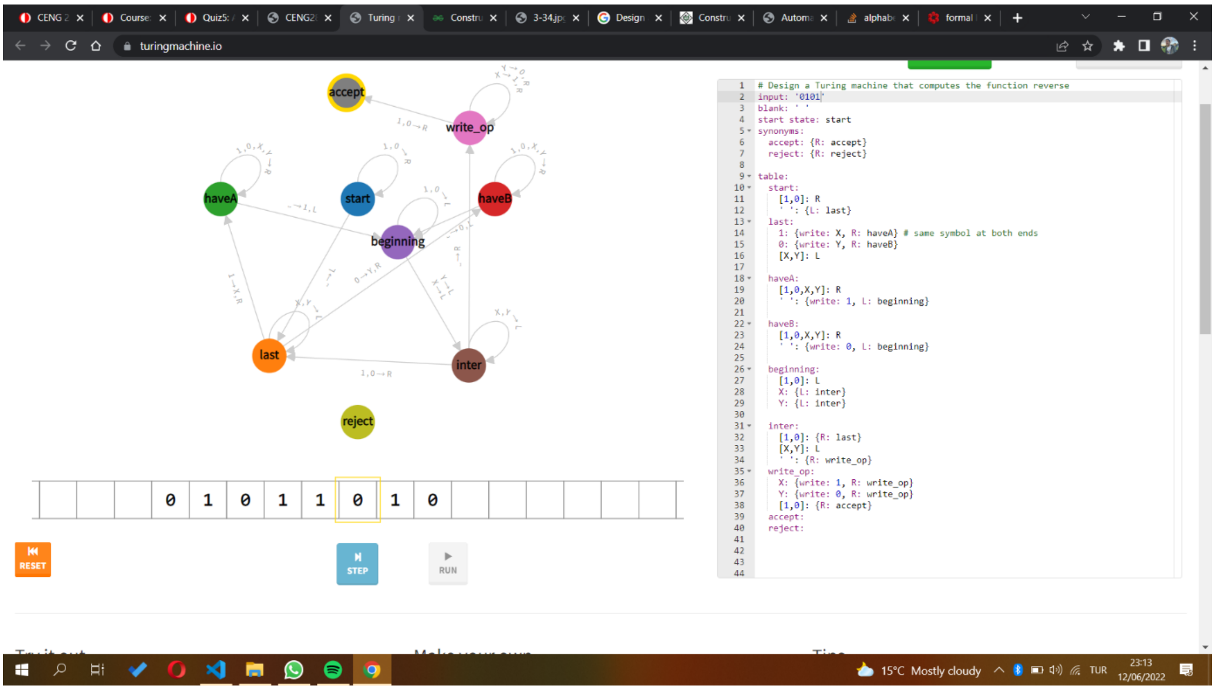
Initial state:1110



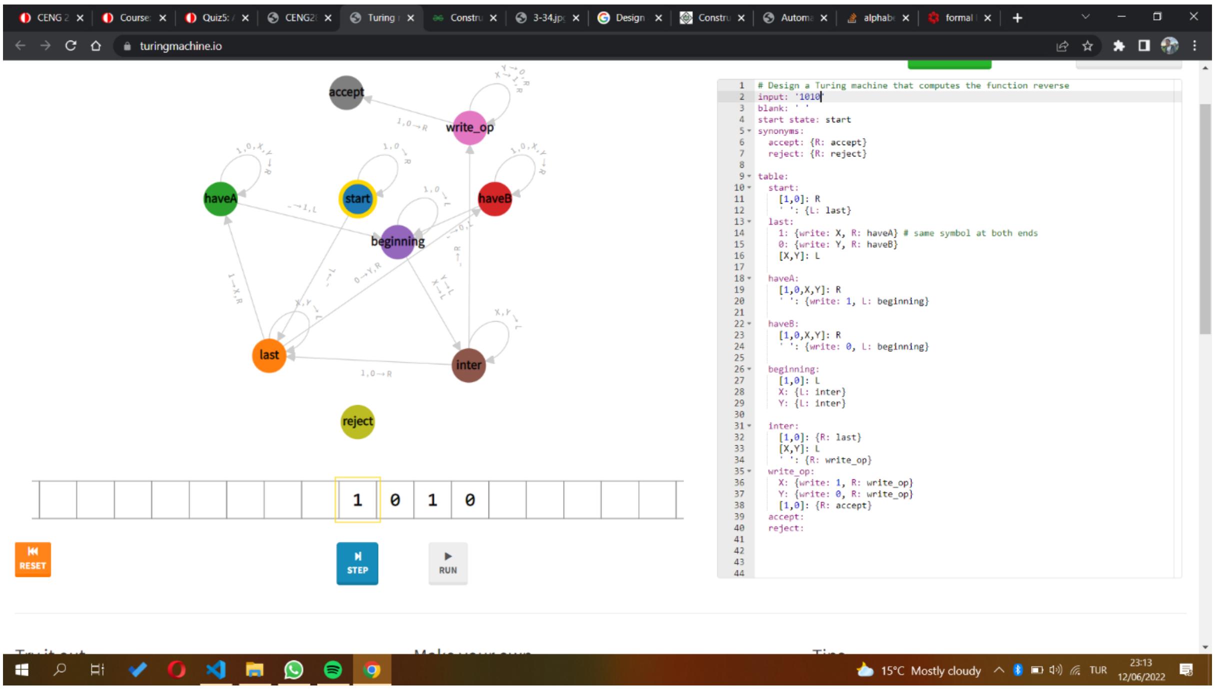
End state: 11100111



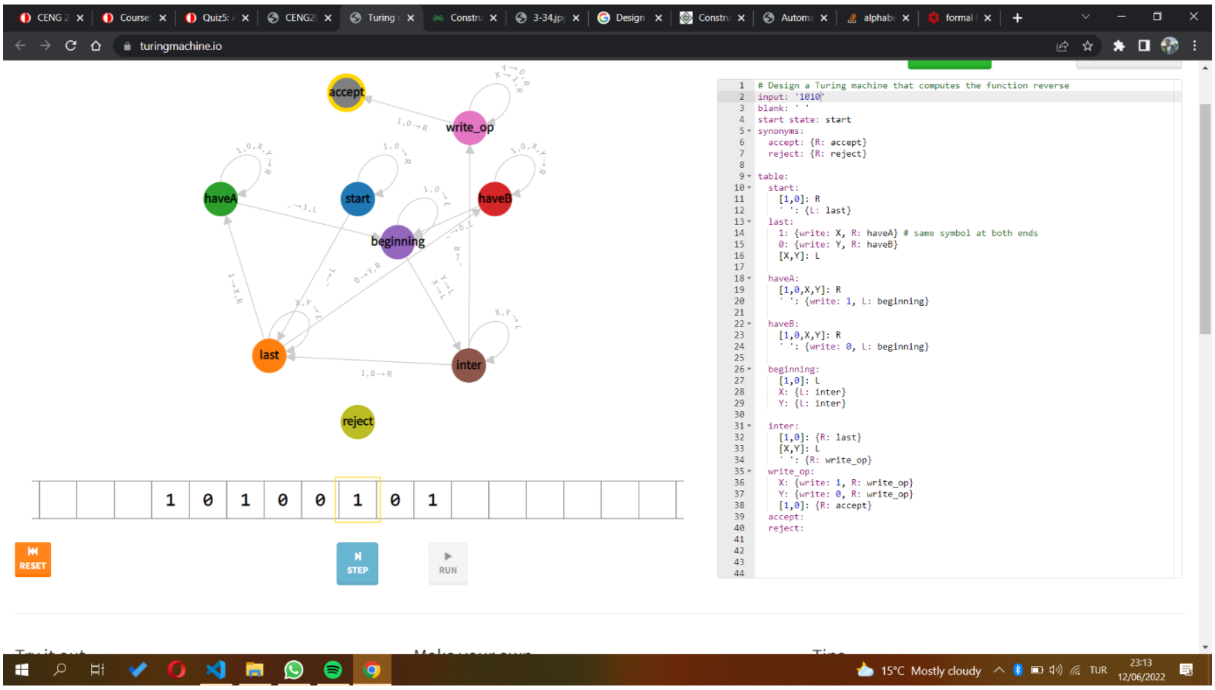
Initial state: 0101



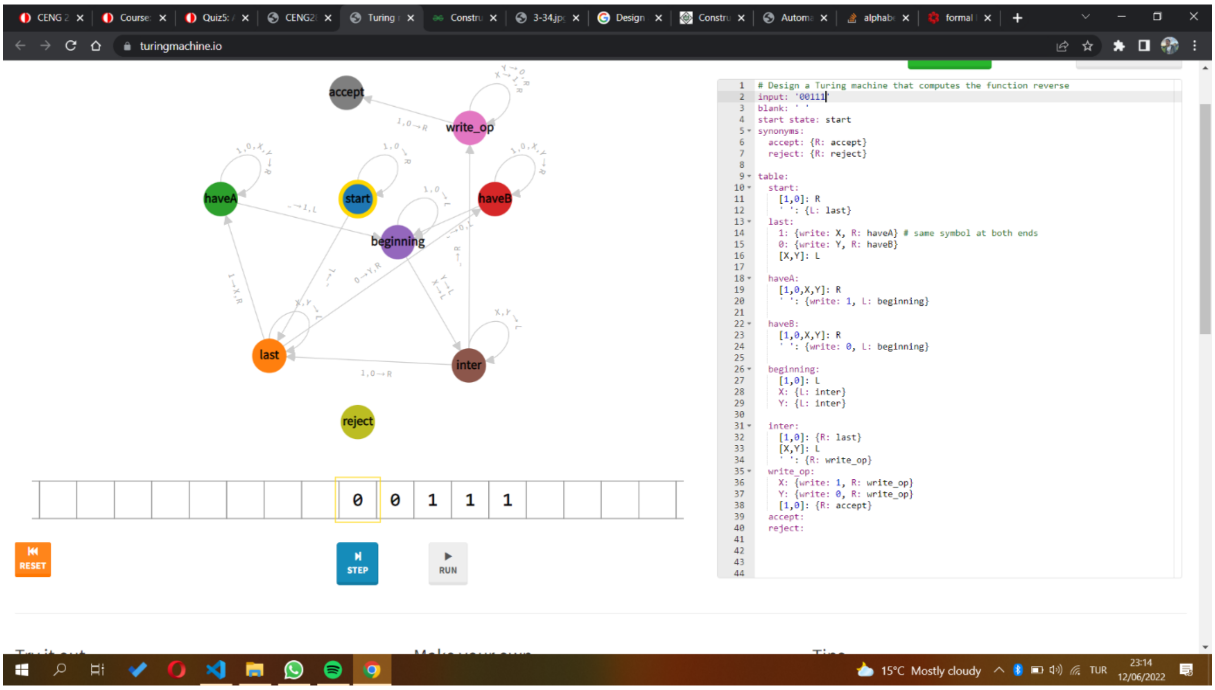
End State: 01011010



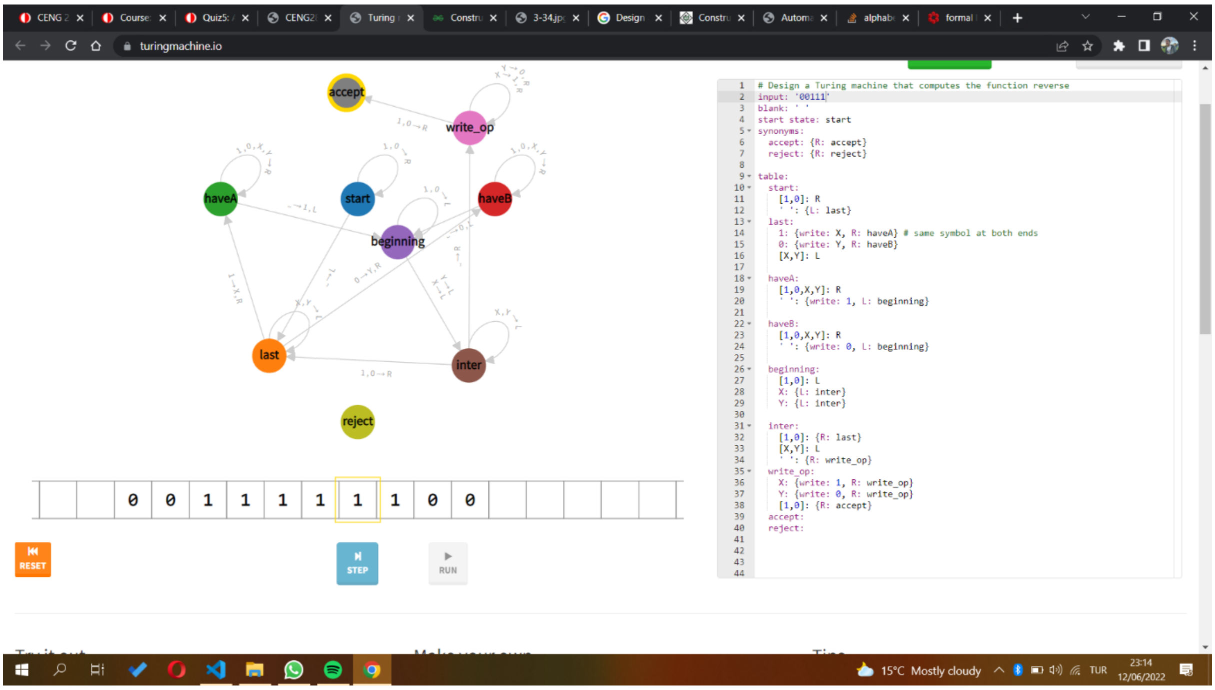
Initial state:1010



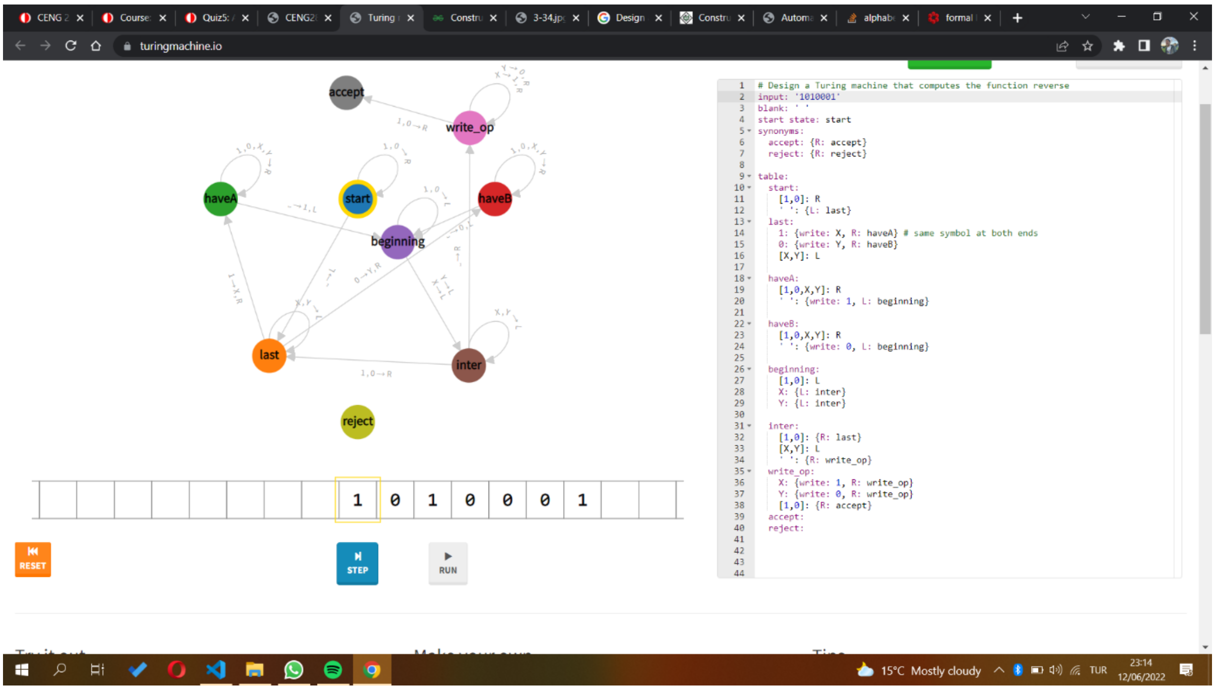
End State:10100101



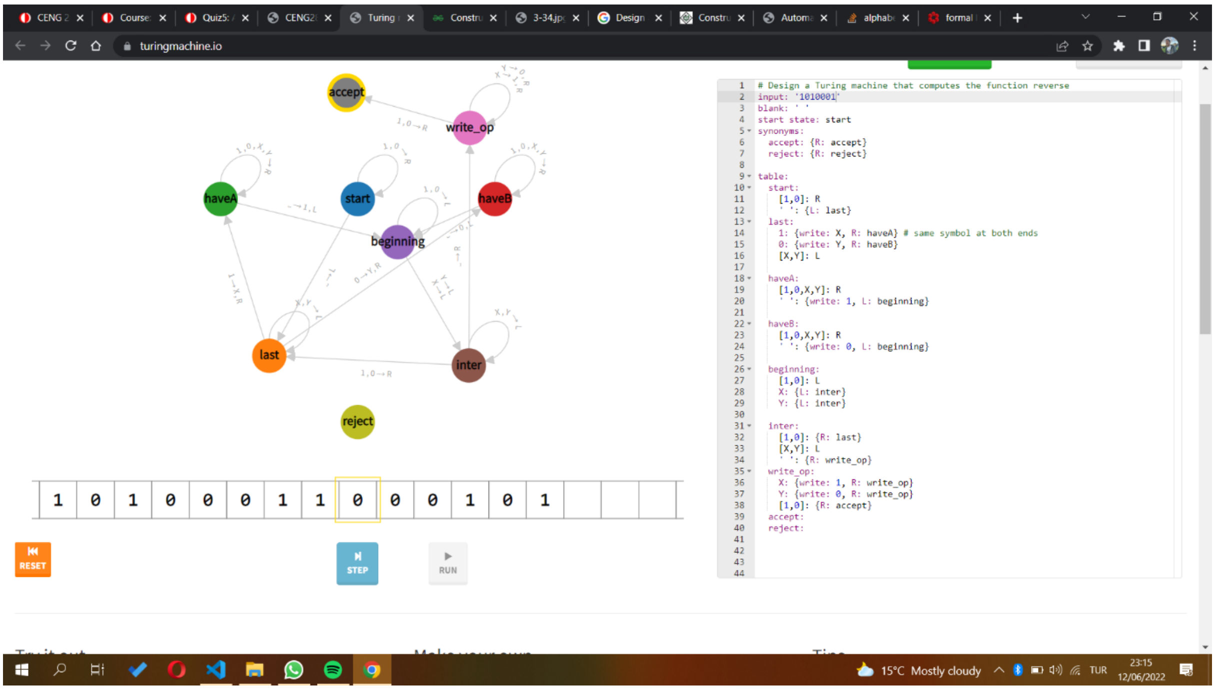
Initial State: 00111



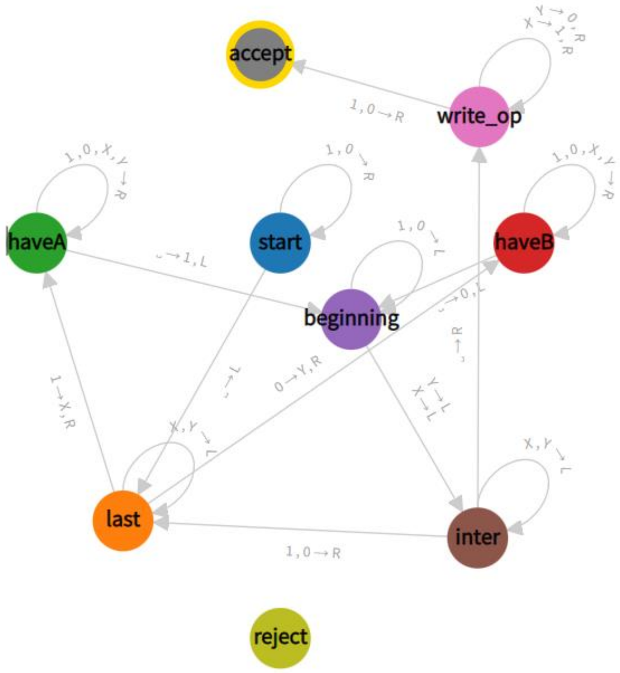
End State: 0011111100



Initial State: 1010001



End State: 10100011000101



Description of states:

Start State:

If symbol 1 or 0 move right, stay in the start sate

If symbol blank move left, go to state last

Last State:

If symbol is 1, replace it by X and move right go to the haveA state

If symbol is 0, replace it by Y and move right go to the haveB state

If symbol X or Y move to the left stay in the last state

haveA state:

If symbol 1,0,X or Y move right, stay in the haveA state

If symbol blank replace it by 1 move left and go to the beginning state

haveB state:

If symbol 1,0,X or Y move right, stay in the haveB state

If symbol blank replace it by 0 move left and go to the beginning state

Beginning state:

If symbol 1 or 0 move left stay in the beginning state

If symbol is X or Y move left go to the inter state

Inter state:

If symbol 1 or 0 move right go tot the last state

If symbol X or Y, move left stay in the inter state

If symbol blank move right go to the write_op state

Write_op state:

If symbol X replace it by 1 move right stay in the write_op state

If symbol Y replace it by 0 move right stay in the write_op state

If symbol 1 or 0 move right, go to the Accept state

Q3)

Turing machine with two dimensional tape, have one finite control, one read-write head and one two dimensional tape. It has top and left ends, and they goes like right and down. The machine has subparts as rows of small squares.

v	v	v	v	....
h	1	2	6	...
h	3	5	8	..
h	4	9	13	...
...	...	...	..	..

This machine is a pentuple M=(K,Σ,δ,s,H),

δ function is from K x Σ to K x (Σ ∪ { →,↓,←,↑}), and δ(q1,⊳)= (q2, →) , δ(q1,Δ)= (q2, ↑) for all q1

Configuration is : K x N x N x T   where S is set of functions form N x N to Σ

Configuration is represented by current state, current head position, list of all non-blank squares on the tape.

(q1,a1,b1,z1) ⊢M (q2,a2,b2,z2) This holds

if δ(q1,z1(a1,b1)) = (q2,#) and one of the below

a1=a2, b1+1= b2, z1=z2, and # = →

a1=a2, b1-1= b2, z1=z2, and # = ←

a1+1=a2, b1=b2, z1=z2, and # = ↑

a1-1 = a2, b1=b2, z1=z2, and # = ↓

a1=a2, b1=b2, z2(a1,b1)= #, z2(a, b)=z1(a, b) for all other pairs (a, b), and # $\notin$ { $\rightarrow, \downarrow, \leftarrow, \uparrow$}

given a string w lets take $z_w \in T$ be a function which z(i+1,1) = w(i) for 0< i $\leq$ |w|, z(0,b) = $\rhd$

for b $\in$ N, z(a,0) = Δ for all a>0, and z(a, b) = " " , in different situation. If machine as two halting states y and n such that for any string w

$(s,1,1,z_w) \vdash^*_M (y,i,j,z')$ or $(s,1,1,z_w) \vdash^*_M (n,i,j,z')$ For such a machine deciding a language is the set of strings for which halts in the y state.

Standard Turing machines can simulate every move of a Turing machine with two dimensional tape. Hence they are at least as powerful as Turing machines with a two dimensional tape.

Since we have two dimensional tape it requires a quadratic time to find the coordinates that we do our operations and we can do our operations in constant time let's say. If we operate t times we will have $O(t^3)$. So we can say that can be simulated by a standard Turing machine in time that is polynomial in t and n.