Extra Credit

Antonio Zea Jr

November 16, 2022

Task:

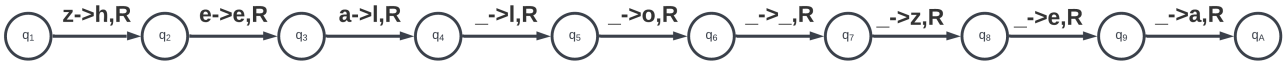
Design a Turing machine that recognizes the language of only one string. This string should be your first name (you can also use a nickname if it's shorter). When the machine accepts it should write on the tape "Hello" in front of the name. For example, if the tape contains "John" after accepting it should contain "Hello John". Implement the machine for the TMSimulator and include the following in your submission: an implementation-level description of the machine, a state diagrams, the code for the TMSimulator, and a copy of the TMSimulator window showing the accepting state.

Implementation-level Description:

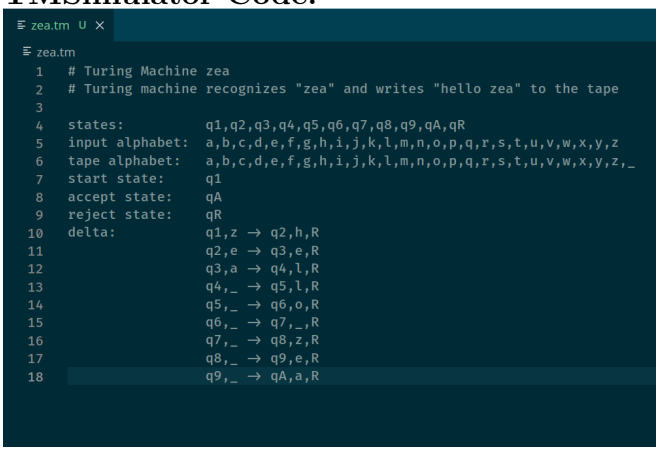
"On input string w :

1. Scan tape for the string "zea␣".
2. As the head scans "zea␣", replace characters with characteres from "hello␣zea␣"
3. Once "hel" has been written, "lo␣zea" gets written onto the blank spaces that were present on the tape after "zea""

State Diagram:



TMSimulator Code:



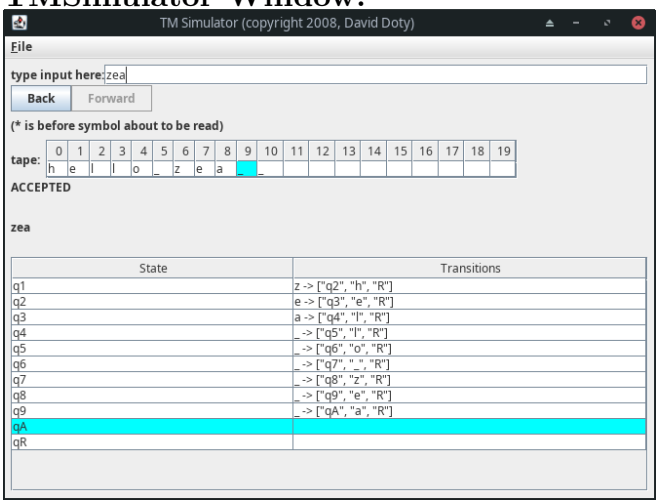
zea.tm U ×

zea.tm

```
# Turing Machine zea
# Turing machine recognizes "zea" and writes "hello zea" to the tape

states:          q1,q2,q3,q4,q5,q6,q7,q8,q9,qA,qR
input alphabet:  a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z
tape alphabet:   a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,_
start state:     q1
accept state:    qA
reject state:    qR
delta:           q1,z → q2,h,R
                 q2,e → q3,e,R
                 q3,a → q4,l,R
                 q4,_ → q5,l,R
                 q5,_ → q6,o,R
                 q6,_ → q7,_,R
                 q7,_ → q8,z,R
                 q8,_ → q9,e,R
                 q9,_ → qA,a,R
```

TMSimulator Window:



TM Simulator (copyright 2008, David Doty)

File

type input here: zea

Back | Forward

(* is before symbol about to be read)

tape:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
	h	e	l	l	o	_	z	e	a	_	_									

ACCEPTED

zea

State	Transitions
q1	z -> ["q2", "h", "R"]
q2	e -> ["q3", "e", "R"]
q3	a -> ["q4", "l", "R"]
q4	_ -> ["q5", "l", "R"]
q5	_ -> ["q6", "o", "R"]
q6	_ -> ["q7", "_", "R"]
q7	_ -> ["q8", "z", "R"]
q8	_ -> ["q9", "e", "R"]
q9	_ -> ["qA", "a", "R"]
qA	
qR	