CSci 435: Formal Languages and Automata

Name: Adam Roy

**Home Assignment 8: Standard Deterministic TM (80 + 10 optional)**

Q1. [20] For a given language below, construct a TM with a *single final state* that accepts it.

1. [10] L = {w ||*w*|is a multiple of 4} where Σ = {*a*, *b*}.

A black background with white text

Description automatically generated

1. [10] L = {w | *anbm an+m* | *n* ≥ 0, *m* ≥ 1} where Σ = {*a*, *b*}.

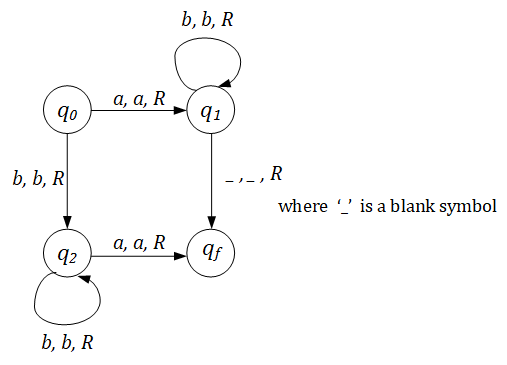
A diagram of a complex flowchart

Description automatically generated

1. [10, optional] L = {w | *anb2n* | *n* ≥ 1} where Σ = {*a, b*}.

Q2. [10] What language is accepted by the Turning machine whose transition graph is in the figure?

Give it in the formal representation, including the regular expression.



The turing machine can be defined as a 7-tuple (Q, Σ, Γ, δ, q0, q𝑓) where:

Q is a finite set of states

Σ is the alphabet

Γ is the tape alphabet

δ is the transition function such that Q x Γ → Q x Γ x {L, R}

q0 is the initial state

q𝑓 is the final and accept state

The language can be defined as L = {w | w = abn + bma | n ≥ 0, m ≥ 1}

Q3. [10] Construct a TM that accepts L = {ww | w ∈ {*a, b*}+ }.

Hint: This is a standard deterministic TM.

So, the TM has to **find** the middle of the string first; then, compare the two halves.

Explain how your TM finds the middle of the string, then give the transition diagram of your TM.

The turing machine works by placing a marker at the beginning of the tape “X” to which it then goes to state q1 where it traverses the rest of the string until it reaches a blank space where it then moves the tape head L and places a marker “Y” which it then finds the middle and proceeds through the states bouncing back and forth between markers accepting the string if each half matches but rejects if any of the strings are not the same or if the string is odd.

A diagram of a algorithm

Description automatically generated

Q4. [20] Construct a TM that computes the following function

1. [10] .

The input *w* is in the unary representation.

A black background with white text and arrows

Description automatically generated

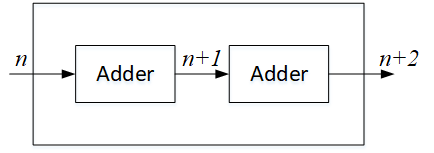
1. [10]  *f*(*x, y*) = *x* + 2*y.*

A diagram of a diagram

Description automatically generated with medium confidence

Q5. [20] Using adders, subtracters, comparers, copiers or multipliers, draw block diagram for TM that compute the functions:

e.g.) *f*(*n*) = *n* + 2



1. [10]*f*(*n*) = *2n.*

A diagram of a mathematical equation

Description automatically generated

1. [10] *f(n) = n!*

A diagram of a block diagram

Description automatically generated