CMSI 3802: Languages & Automata
Homework 4

[Ryan Ramsdell (980388983)]

April 16, 2024

Problem 1

(a) Language Theory

It deals with the expression of computation and information, analyzing how alphabets can compute functions.

(b) Automata Theory

It analyzes specific, formal models of computers, including things like the turing machine, pascal's adder, and more.

(c) Computability Theory

It is the discussion of whether or not every possible function can be computed, the answer being "no"; specificially, a function that knows if a given program will terminate.

(d) Complexity Theory

Complexity theory is similar to computability theory, but specifically with reguard to how long a function will take to compute.

Problem 2

(a) $L_1 \cup L_2$

$\{1, 011, 10, 1\}$

(b) $L_1 \cap L_2$

$\{10\}$

(c) L_1L_2

$\{010, 01, 01110, 0111, 1010, 101\}$

(d) L_2^*

$\{\epsilon, 10, 1, 11, 101, 110, 1010, ...\}$

Problem 3

(a) The empty language

$S \to (norules)$

(b) $\{0^i1^j2^k \mid i = j \vee j = k\}$

$$\begin{array}{ll} S \to 0S1 \mid A & \\ A \to 1A2 \mid B & (\text{for } i = j) \\ B \to 0B \mid 2B \mid \epsilon & (\text{for } j = k) \end{array}$$

(c) $\{w \in \{0,1\}^* \mid w \text{ does not contain the substring } 000\}$

$$S \to 0S \mid 01S \mid 001S \mid 1S \mid \epsilon$$

(d) $\{w \in \{a,b\}^* \mid w \text{ has twice as many } a\text{'s as } b\text{'s}\}$

$$S \to aSaSb \mid ab \mid \epsilon$$

(e) $\{a^nb^na^nb^n \mid n \geq 0\}$

$$\begin{array}{l} S \to aSb \mid T \\ T \to aTb \mid \epsilon \end{array}$$

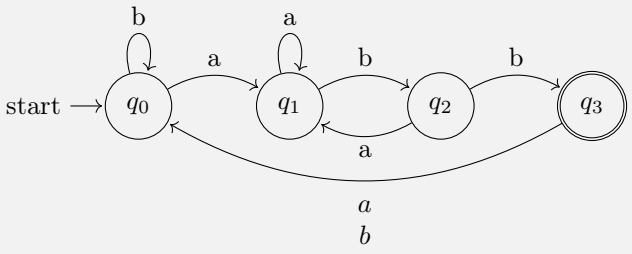
Problem 4

$$V = \{n, f, e, d, D, E, E', F\}$$
$$\Sigma = \{\text{"0"}, \text{"1"}, \text{"2"}, \text{"3"}, \text{"4"}, \text{"5"}, \text{"6"}, \text{"7"}, \text{"8"}, \text{"9"}, \text{"."}, \text{"E"}, \text{"e"}, \text{"+"}, \text{"–"}\}$$
$$R = \left\{ \begin{array}{l} n \to dD \mid dDF \mid dDE \mid dDFE \\ D \to dD \mid d \\ F \to \text{"."}dD \\ E \to \text{"E"}E' \mid \text{"e"}E' \\ E' \to dD \mid \text{"+"}dD \mid \text{"-"}dD \\ d \to \text{"0"}..\text{"9"} \end{array} \right\}$$
$$S = n$$

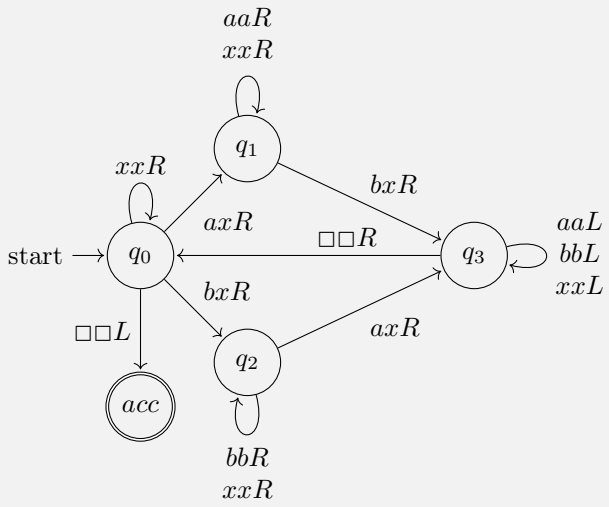
Problem 5

Give Turing Machines that recognize the following languages. If any of the languages below are Type-3, you may (and are encouraged to) give a FA in lieu of a TM recognizer, if the FA is simpler.

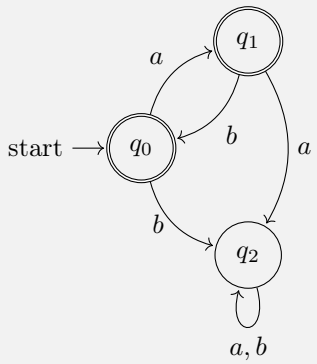
(a) $\{w \in \{a, b\}* \mid w \text{ ends with } abb\}$



(b) $\{w \in \{a, b\}^* \mid \#_a(w) = \#_b(w)\}$ (same number of a's and b's)

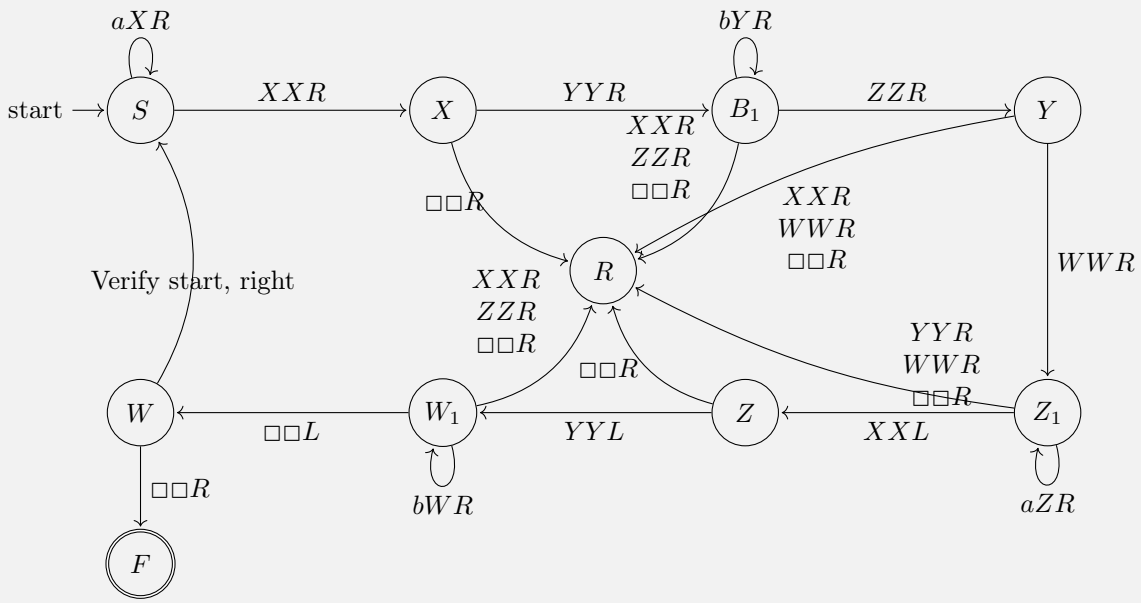


(c) $\{w \in \{a, b\}* \mid w$ alternates a's and b's$\}$



(d) $\{a^n b^n a^n b^n \mid n \geq 0\}$

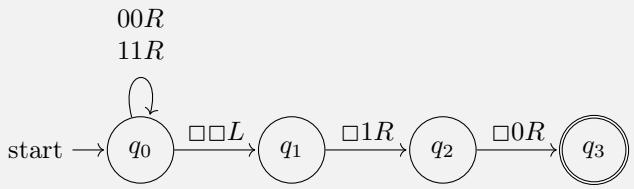
Note: Apologies for the messy diagram, I spent like 45 minutes on this one trying to get LaTeXto format it nicely, and I decided to just take the L.



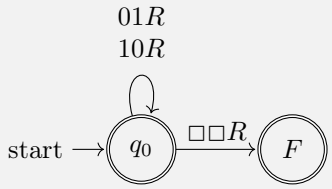
Problem 6

Give Turing Machines that compute the following functions, where the input and output are binary numerals.

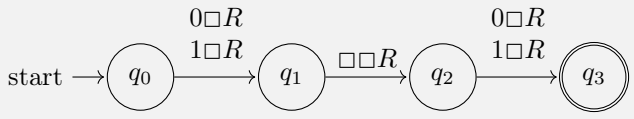
(a) $\lambda n.2n + 2$



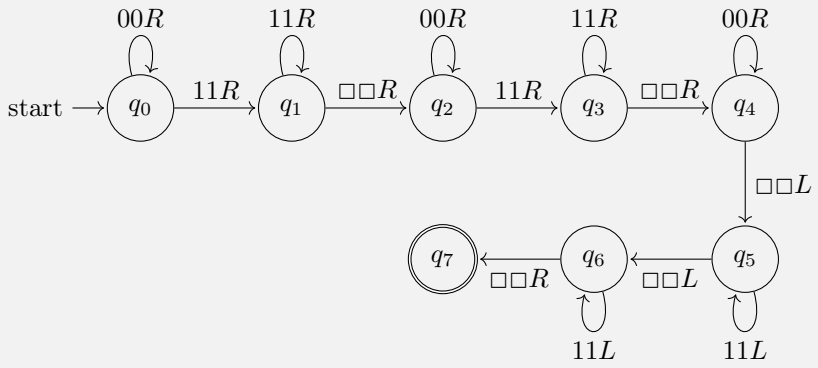
(b) one's complement



(c) The function described in Python as `lambda n: str(n)[1:-1]`



(d) Maximum bit-string length of two numerals, after leading zeros are removed, where the input is the two numerals separated by a single blank



Problem 6

For the JavaScript/Python expression `5 * 3 - 1 ** 3`,

(a) Show a 3AC machine program to evaluate this expression, leaving the result in r_0

```
COPY 5, r1
COPY 3, r2
MUL r1, r2, r3

COPY 1, r4
COPY 3, r5
POW r4, r5, r6

SUB r3, r6, r0

WRITE r0
HALT
```

(b) Show a Stack machine program to evaluate this expression, leaving the result on the top of the stack.

```
PUSH 5
PUSH 3
MULT
PUSH 1
PUSH 3
POW
SUB
```

Problem 8

Characterize each of the following languages as either (a) regular, (b) context-free but not regular, (c) recursive but not context-free, (d) recursively enumerable but not recursive, or (e) not even recursively enumerable.

(a) $\{a^i b^j c^k \mid i > j > k\}$

c: recursive but not context-free

(b) $\{a^i b^j c^k \mid i > j \wedge k \leq i - j\}$

c: recursive but not context-free

(c) $\{\hat{M}w \mid M \text{ accepts } w\}$

d: recursively enumerable but not recursive

(d) $\{G \mid G \text{ is context-free} \wedge L(G) = \varnothing\}$

c: recursive but not context-free

(e) $\{a, b\}^*\{b\}^+$

a: regular

(f) $\{\hat{M} \mid M \text{ does not halt }\}$

e: not even recursively enumerable

(g) $\{w \mid w \text{ is a decimal numeral divisible by } 7\}$

a: regular

(h) $\{www \mid w \text{ is a string over the Unicode alphabet}\}$

c: recursive but not context-free