Homework 5
CSC 445-01: Theory of Computation

Matthew Mabrey, Luke Kurlandski

April 8, 2021

2.30

We will prove L is not a context free language using a proof by contradiction and the Pumping Lemma.

$$L = \{t_1\#t_2\#...\#t_k | k \geq 2, t_i \in \{a, b\}^*, \exists(t_i = t_j, i \neq j)\}$$

Suppose L is a CFL. Then for a string $s \in L$ of length greater than p, there exists some decomposition of s, $s = uvxyz$ such that

1. $uv^ixy^iz \in L$ for $i \geq 0$
2. $|vy| > 0$
3. $|vxy| \leq p$

We let $s =$ because $|s| \geq p + 1$ and $s \in L$.

Ooooof

3.8

b

On input string w

1. Move the head to the front of the tape. Scan the tape and mark the first 1 that has not been marked. If none are found, move to stage 5.
2. Move the head to the front of the tape. Scan the tape and mark the first 0 that has not been marked. If no 0 is found, reject.
3. Scan the tape and mark the first 0 that has not been marked. If no 0 is found, reject.
4. Go to stage 1
5. Move the head to the front of the tape. Scan the tape to see if any unmarked 0s remain. If none remain, accept, else reject.

c

On input string w

1. Move the head to the front of the tape. Scan the tape and mark the first 1 that has not been marked. If none are found, move to stage 5.
2. Move the head to the front of the tape. Scan the tape and mark the first 0 that has not been marked. If no 0 is found, accept.
3. Scan the tape and mark the first 0 that has not been marked. If no 0 is found, accept.
4. Go to stage 1
5. Move the head to the front of the tape. Scan the tape to see if any unmarked 1s remain. If none remain, reject, else accept.

3.15

b

Suppose we have Turing Machines M_1 and M_2 that decide languages L_1 and L_2 respectively. We describe the Turing Machine M' that decides the concatenation of L_1 and L_2.

On input string w

1. Run M_1 on the first portion of w
   - If M_1 enters an accept state w, begin stage 2
   - If M_1 does not accept, reject
2. Run M_2 on the remaining portion of w.
   - If M_2 accepts, then accept
   - Else, reject

c

Suppose we have a Turing Machine M that decides the language L. We describe the Turing Machine M' that decides the language L^*.

On input string w

1. Run M on w.
   - If w is empty, accept
   - If M enters an accept state, repeat stage 1 at the current point on input string

d

Suppose we have a Turing Machine M that decides the language L. We describe the Turing Machine M' that decides the language L^c.

On input string w

1. Run M on w.
   - If M accepts, then reject
   - Else accept

e

Suppose we have Turing Machines M_1 and M_2 that decide languages L_1 and L_2 respectively. We describe the Turing Machine M' that decides the intersection of L_1 and L_2.

On input string w

1. Run M_1 on w
   - If M_1 accepts, continue to stage 2
   - Else reject
2. Run M_2 on w
   - If M_2 accepts, accept
   - Else reject