

CS 475: Formal Models of Computation

Homework 4

1. [10 points] Consider a variant of a PDA, called a *queue-machine*, which is an NFA with a queue instead of a stack. A queue is a data structure which allows reading and popping the symbol at the front of the queue, or pushing a symbol to the end of the queue (i.e., a first in, first out model). A queue-machine may not always halt for some inputs. Give a formal definition of what a queue machine is, and then argue that for every TM, there is a queue-machine that simulates it.

Answer (sketch). There are several ways in which we can prove that a Turing Machine can be simulated by a queue-machine. For example, in class we saw that a PDA with two stacks can simulate a T.M.. We can simulate a queue with two stacks, by using one stack as the front of the queue (pop operation), and the other as the back of the queue (push operation). When the first stack becomes empty, we copy the contents of the second stack into the first one.

Alternatively, we can choose two distinctive characters, say \$ and ϕ , to help us simulate the left and right moves of a T.M. tape. When the computation starts, we push ϕ into the queue, and then push the whole input string into the queue. The symbol ϕ signals the leftmost position of a T.M. tape. For a right move, pop the top of the queue, and push the character that would have been written in the tape (with special considerations for never removing ϕ from the queue). For a left move, push \$, and keep popping and pushing characters until \$ is found. The previous character popped was the character to the *left* of the original position.

2. [10 points] Let $\log^*(x) = n$, for $x, n \in \mathbb{N}$, be the number of times \log should be successively applied to x to obtain 0. That is:

$$\underbrace{\log(\log(\log(\dots \log(x) \dots)))}_{\log^*(x)=n \text{ times}} = 1.$$

Describe a Turing Machine that computes $\log^*(x)$. Assume that the input to the Turing Machine is a string of the form 1^x .

Answer (sketch). First we need a method to compute $\log(x)$. Consider a T.M. with two tapes. In one tape, the input tape, we write x 1s, and slash every other 1, from left to right, and we repeat the process until all characters are slashed. We use the second tape to count the number of times we go from left to right slashing ones. When all 1s are slashed, the second tape contains $\log[x]$.

Now, we can copy the contents of the second tape to the first tape, and use a third tape to count the number of times we applied \log .

3. [10 points] Prove that the class of recognizable languages is closed under Kleene $*$. (Hint: Think nondeterministically.)

Answer. Let A be a recognizable language, with $M = L(A)$. To prove that A^* , we construct a nondeterministic T.M. M' that recognizes it. For M' , on input w , M' partitions it (nondeterministically) into $w = xy$. Note that both x and y may be the empty string. The machine M' accepts w if it is the empty string, if M accepts w , or if, recursively, M' accepts both x and y .

4. [10 points] For this problem, consider the Turing Machine model in which the machine can move its head to the left, to the right, or to stay in the same tape cell. Consider the following language:

$$A = \{\langle M\#w \rangle \mid M \text{ moves its head to the left at least once on input } w\}$$

Is A recognizable? Is A decidable?

Answer (sketch). The language A is decidable. Consider a T.M. T that decides it. The machine T has three tapes: tape 1 is the regular input/work tape; on tape 2 it simulates M on w ; and on tape 3 it records information about the transitions of M .

$T =$ On input $\langle M\#w \rangle$, simulate M on w .

- If M moves its head to the left, accept.
- If M moves its head to the right, to a position on the tape not occupied by w (a blank position), record on tape 3 the pair (q, ϵ) , in which q is the state of M when it moved its head to the right.
- If M is in a position of the tape not occupied by w , and the head does not move in the transition, then record on tape 3 the pair (q, c) , in which q is the previous state of M , and $c \in \Gamma$, is the character written on the tape before the current transition.
- Reject if a pair (q, ϵ) or (q, c) is written more than once on tape 3, or if M decides w without moving its head to the left.

Since M is deterministic, when a pair such as (q, ϵ) or (q, c) is written twice on the tape, M has entered a cycle that does not have any left moves. Note that there are finitely many such pairs.