CSE 540 Wk1

Wednesday 28th August, 2024

In the last lecture we visited the motivation behind complexity theory. In this lecture we will introduce the basic idea and intuition for Turing machines.

0.1 Why Turing machines

In the undergraduate level computation class, we may have already studied multiple computation models including Finite State Machines (FSMs), Push Down Automatas (PDAs), and maybe some introductions on Turing machines. A natrual question to ask even before studying the Turing machine is: Why is Turing machine needed? And what tasks we can perform with a turing machine but not with a FSM or PDA?

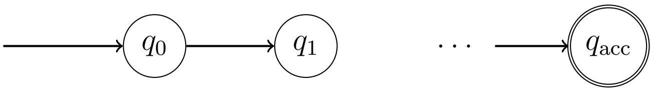

Figure 1: A Finite State Machine

The intuition is, with FSMs, we can do certain computations, maybe decide a regular language. But it can not do much more than that (for example, it won't be able to recognize the language $\left\{0^k1^k \mid k \in \mathbb{N}\right\}$).

So people came up with another model, adding a stack to the finite state machine, but it has its limitations (for example, it won't be able to recognize the language $\left\{0^k1^k0^k \mid k \in \mathbb{N}\right\}$)

Both Finite State Machines and Push Down Automatas has certain capability, and some "built-in" memory (states, stack), however their memory is limited. For example, finite state machines can not "remember" how many 0's it has seen and match that with the following 1's, and similarly even if we give it a stack it can only sort of memorize one thing at a time. Thus, we now introduce a machine that can memorize ...

0.2 Introduction to Turing Machines

Imagine the tape of a cassette, for Finite Automatas, the input goes one-way. For example, given $(000111) \in \{0,1\}^*$, and a finite automata F, after reading the first 3 zeros, it can't not go back and read the zeros again. On the other hand, in a turing machines, we give it the power, letting the head pointer for the input tape to move freely so that it could read the zeros again.

Definition 0.1 (Turing Machine (informal))**.** A Turing Machine M

0.3 A few examples regarding Turing Machines

0.4 Why tape size doesn't matter, and Oblivious Turing Machines

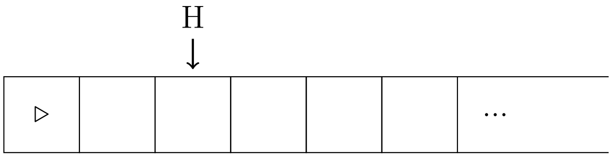

Figure 2: a turing machine tape