CSc 30400 Introduction to Theory of Computer Science
8th Homework Set

1. Run the machine that partially computes the function $f(n) = \log n$ from the slides for the following input numbers:
   (a) 0 (input ε, machine should loop)
   (b) 1 (input 1, machine should halt and output 0)
   (c) 2 (input 11, machine should halt and output 1)
   (d) 3 (input 111, machine should loop)
   (e) 6 (input 111111, machine should loop)
   (f) 8 (input 11111111, machine should halt and output 3)

   This question is time consuming. If you understand how Turing Machines work you don't have to do it. However, running it will help you design a TM for language L_2 (see question 3 below). Also make sure that you understand what a partially computable function is ($\log n$ is not defined for $n = 0$, 3 or 6, so the machine shouldn't give an output when the input is one of these numbers, it should loop for ever).

2. (question 4 from set 7). In many cases as we saw in class, it is useful when designing a Turing machine to have a special symbol (for example a \$) to denote the start (left end) of the tape (just like the \$ symbol we used in PDAs to denote the end of the stack). By using this symbol you can easily move between states (if you are in state q_i and see the \$ symbol then move to state q_j). Design a transition diagram that shifts the input by one block to the right and places a \$ in the leftmost position and accepts when the process is done (of course you could do this by moving one block to the left and placing the \$ there but I don't want to see this solution...)

   Example: $\boxed{0}\boxed{1}\boxed{0}\boxed{0}\boxed{1}\boxed{\ }\cdots$, should produce $\boxed{\$}\boxed{0}\boxed{1}\boxed{0}\boxed{0}\boxed{1}\boxed{\ }\cdots$

3. (question 5 from set 7) Design Turing Machines that accept the languages $L_1 - L_4$ of questions 1 and 2 of set 7. Make sure that you also give a "high level program" for every case and not only the transition diagram of the machine. In all parts the alphabet Σ contains only the symbols that are used each time in the language (for example, for L_3: $\Sigma = \{0, 1, \#\}$). You are free to define the alphabet Γ of the tape to be whatever you want.

4. Show that the following functions are computable (design Turing machines that compute them)

   (a) $f : \mathbb{N} \to \mathbb{N}, f(n) = 3n + 2$

   * (b) $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}, f(x, y) = x \cdot y$

5. Show that $f : \mathbb{N} \to \mathbb{N}$ with

$$f(n) = \begin{cases} n - 3, & n \geq 3 \\ \uparrow, & \text{else} \end{cases}$$

   is partially computable (design a Turing machine that partially computes it). The $\uparrow$ symbol denotes that f is undefined for values 0,1 and 2.

* 6. The task in this question is to prove that Computable functions are closed under composition. In other words you should prove that, if two functions $f : \mathbb{N} \to \mathbb{N}$ and $g : \mathbb{N} \to \mathbb{N}$ are computable then $f \circ g : \mathbb{N} \to \mathbb{N}$ is also computable.

   (a) We said that when we design a machine with output we only require the output to be consecutive but we don't require the head to point to the leftmost symbol. In this part, you have to fix this. Take an arbitrary Turing machine with output. When it reaches the final state it has a bunch of consecutive 1s on the tape but the head doesn't point to the first one. Make the final state non final (think of the final state being your new start state) and try to write some more transitions in order to make sure that when the machine halts the head is in the first 1 of the output.

   Hint: There are three possibilities:

   - The head points to some symbol of the output
   - The head points to a blank box somewhere in the left of the output
   - The head points to a blank box somewhere in the right of the output

   Your transition should take care of all those situations. For the first one you should go left until you find the beginning of the output (a blank box). Taking care of rest two would also be a simple task if you knew whether you were on the right or on the left side of the output (just move towards the correct direction until you find the output) but this is not the case. Moving in the

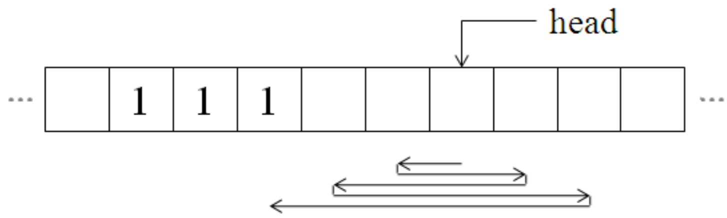

Figure 1: Zig-zag move in order to reach the output

wrong direction would make the machine loop for ever. Make a zig zag search like the one shown in the figure. You should place several mark signs (for example \$) in order to do so (place a mark sign when you see a new blank box and then change direction). When you find the output you have to remove the extra mark signs you put.

(b) If f and g are computable then there are TMs M_1 and M_2 that compute them. We would like to compute $f(g(x))$. Running M_2 on input x would give us $g(x)$. Part (a) of this question ensures us that the output that we are taking from M_2 is of the form "consecutive 1s and the head points to the first symbol of the string", so it can serve as an input to M_1 as is. Can you combine M_1 and M_2 in such a way that $f \circ g$ is computed?