Homework 5 – Due Tuesday, October 20th, 2020 before 2:00PM

Reminder Collaboration is permitted, but you must write the solutions *by yourself without assistance*, and be ready to explain them orally to the course staff if asked. You must also identify your collaborators. Getting solutions from outside sources such as the Web or students not enrolled in the class is strictly forbidden.

Exercises Please practice on exercises and solved problems in Chapter 3. The material they cover may appear on exams.

Problems There are 2 mandatory problems and one bonus problem.

Note: This homework contains some "Turing machine programming" questions using a web-based platform for simulating the execution of TMs. You might wonder how come there are programming assignments in a "theory of computation" class. Well, the reason is that this is the best way to understand the model, its generality, and its limitations. Rest assured that this is going to be your only for-submission programming assignment in this class!

1. (**Reading TM descriptions**)

   (a) (**20 points**) Give a state diagram of the TM whose implementation level description is below.

   "On input w

   1. If the first symbol is blank then *accept*. If it is b then *reject*. If it is a then erase this a and go to the right and go to the next step.

   2. Move the head on the tape right until the blank symbol is found. After it is found move the head to the left (to the last symbol of the string) and go to the next step.

   3. If the last symbol is not b then *reject*. Otherwise, erase this b and go to the left and go to the next step.

   4. Move the head on the tape left until the blank symbol is found. After it is found move the head to the right (to the first symbol of the string) and go back to step 1."

   (b) (**20 points**) What language is decided by the TM from (c)? Is it context-free? Justify your answer. (Recall that to justify that the language of a TM is L, you need to show two things: (a) that any word x that the TM accepts in L, and (b) that if $w \in L$ then the TM accepts w.)

2. (**Programming TMs**) Write Turing machines that decide the following languages. That is, the machines should always halt after a finite number of steps, and accept an input x if and only if x is in the language. Implement your TMs in the following environment: `http://morphett.info/turing/turing.html`. That is, your solution should contain:
   (a) An English, intermediate-level explanation of your code.

(b) Code that can be run directly on that website. (Please add comments and make it as readable as possible.)
(c) Transcripts of executions of your code on the test inputs provided below.
(Note that your solution will be run on additional test inputs.)

(a) (**30 points**) $\{w \in \{0,1\}^{2n} \mid$ the nth symbol in w is 0 $\}$. Test inputs: 01, 1100010, 110011.

(b) (**30 points**) $\{ww | w \in \{0,1\}^*\}$. Test inputs: 11, 101, 1011.

3. (**Bonus**) Given any CFG $G = (V, \Sigma, R, S)$, describe a TM M that *recognizes* $L(G) \subseteq \Sigma^*$. That is, for each $x \in \Sigma^*$, $M(x)$ halts in the accept state if and only if $x \in L(G)$. (If $x \notin L(G)$ then $M(x)$ may either halt in the reject state or run indefinitely.)
Think but not turn in: Can you design a TM that *decides* this language, namely a TM that recognizes $L(G)$ and also halts on all inputs after a finite number of steps?