

## CS 121 Homework 4: Fall 2020

**Some policies:** (See the course policy at <http://madhu.seas.harvard.edu/courses/Fall2020/policy.html> for the full policies.)

- **Collaboration:** You can collaborate with other students that are currently enrolled in this course (or, in the case of homework zero, planning to enroll in this course) in brainstorming and thinking through approaches to solutions but you should write the solutions on your own and cannot share them with other students.
- **Owning your solution:** Always make sure that you “own” your solutions to this other problem sets. That is, you should always first grapple with the problems on your own, and even if you participate in brainstorming sessions, make sure that you completely understand the ideas and details underlying the solution. This is in your interest as it ensures you have a solid understanding of the course material, and will help in the midterms and final. Getting 80% of the problem set questions right on your own will be much better to both your understanding than getting 100% of the questions through gathering hints from others without true understanding.
- **Serious violations:** Sharing questions or solutions with anyone outside this course, including posting on outside websites, is a violation of the honor code policy. Collaborating with anyone except students currently taking this course or using material from past years from this or other courses is a violation of the honor code policy.
- **Submission Format:** The submitted PDF should be typed and in the same format and pagination as ours. Please include the text of the problems and write **Solution X:** before your solution. Please mark in Gradescope the pages where the solution to each question appears. Points will be deducted if you submit in a different format.
- **Late Day Policy:** To give students some flexibility to manage your schedule, you are allowed a net total of **eight** late days through the semester, but you may not take more than **two** late days on any single problem set. No exceptions to this policy.

**By writing my name here I affirm that I am aware of all policies and abided by them while working on this problem set:**

**Your name:** (Write name and HUID here)

**Collaborators:** (List here names of anyone you discussed problems or ideas for solutions with)

**No. of late days used on previous psets (not including Homework Zero):**

**No. of late days used after including this pset:**

## Questions

Please solve the following problems. Some of these might be harder than the others, so don't despair if they require more time to think or you

bonus questions if you have the time to do so. If you don't have a proof for a certain statement, be upfront about it. You can always explain clearly what you are able to prove and the point at which you were stuck. You can always simply write **"I don't know"** and you will get 15 percent of the credit for this problem. If you are stuck on this problem set, you can use Piazza to send a private message to all staff.

**Note on reading the textbook:** If you are stuck on some of the problems, try consulting the book to 1) understand the concepts the question is referencing, and 2) review the way similar theorems are proved in the book.

**Problem 0 (5 points):** True or False: I have completed the midterm feedback survey. (True worth 5 points, False, or I don't know worth 0 points.)

**Solution 0:**

**Problem 1 (15 points):** Let  $T$  be the Turing Machine with alphabet  $\Sigma = \{\triangleright, 0, 1, \phi\}$  and transition function  $\delta : [1] \times \Sigma \rightarrow [1] \times \Sigma \times \{L, R, S, H\}$  given by  $\delta(0, \triangleright) = \text{invalid}$ ,  $\delta(0, 0) = (-, 1, H)$ ,  $\delta(0, 1) = (0, 0, R)$  and  $\delta(0, \phi) = (-, 1, H)$ . Describe the function computed by  $T$ . (Hint: two cases that you may want to consider: (1) if the input and output of  $T$  are interpreted as integers written with least significant digit first (so  $(x_0 \cdots x_{n-1})$  represents the integer  $X = \sum_{i=0}^{n-1} x_i 2^i$ ), what is the output of  $T$  on input 121? (2) What is the output of  $T$  on input 000?)

**Solution 1:**

**Problem 2:** In this problem we work with functions over a ternary alphabet, i.e.,  $f : \{0, 1, \#\}^* \rightarrow \{0, 1, \#\}^*$ . What this means is that

1. An  $\#$  symbol on the tape is a valid input symbol, and
2. The output of the machine is the concatenation of all the symbols from  $\{0, 1, \#\}$  on the tape when the machine halts.

(Outside this problem, we normally use  $\#$  as a blank character which is not part of the output.)

**Problem 2.1 (25 points):** Give a Turing Machine with alphabet  $\Sigma = \{\triangleright, 0, 1, \#, \phi\}$  to compute the function  $\text{Clean}_{\#} : \{0, 1, \#\}^* \rightarrow \{0, 1, \#\}^*$  which "erases" all the  $\#$ 's on the tape. Specifically if  $x_0, \dots, x_{n-1} \in \{0, 1, \#\}$  and  $0 \leq i_0 < i_1 < \dots < i_{k-1} < n$  are indices such that for all  $j \in [k]$ ,  $x_{i_j} \in \{0, 1\}$  and for all  $\ell \notin \{i_0, \dots, i_{k-1}\}$ ,  $x_{\ell} = \#$ , then  $\text{Clean}_{\#}(x_0, \dots, x_{n-1}) = x_{i_0} \cdots x_{i_{k-1}}$ .

**Solution 2.1:**

**Problem 2.2 (15 points):** Let  $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$  be computed by a Turing Machine  $M$  with  $k$  states. Let  $g : \{0, 1\}^* \rightarrow \{0, 1\}^*$  be computed by a Turing Machine  $N$  with  $\ell$  states. Show that the function  $h(x) = g(f(x))$  can be computed by a Turing Machine with  $k + \ell + 1000000$  states.

**Solution 2.2:**

**Problem 2.3 (10 points):** Assume there exists a function  $U : \{0,1\}^* \rightarrow \{0,1\}$  that is not computable. Let  $f : \{0,1\}^* \rightarrow \{0,1\}^*$  be a computable function. For  $g : \{0,1\}^* \rightarrow \{0,1\}^*$ , let  $h(x) = g(f(x))$ . Prove or disprove the following statements:

1. For all such  $f$ ,  $g$ , and  $h$ , if  $g$  is not computable, then  $h$  is not computable.
2. For all such  $f$ ,  $g$ , and  $h$ , if  $h$  is not computable, then  $g$  is not computable.

**Solution 2.3:**

**Problem 3:** In this multi-part question you will construct a Turing Machine to multiply integers. It will be convenient to work with functions over a ternary alphabet, i.e.,  $f : \{0,1,@\}^* \rightarrow \{0,1,@\}^*$ . What this means is that

1. An @ symbol on the tape is a valid input symbol, and
2. The output of the machine is the concatenation of all the symbols from  $\{0,1,@\}$  on the tape when the machine halts.

We will also view strings in  $\{0,1\}^*$  as non-negative integers with least significant bit first. (E.g. we equate the string 011 with the integer 6 etc., and integer operations on strings will mean “Take the integer corresponding to this string, perform the given operation, and then reconvert the result to a string”.)

**Problem 3.1 (15 points):** Let  $f_1 : \{0,1,@\}^* \rightarrow \{0,1,@\}^*$  be the partial function given by  $f_1(x@y@z) = (x-1)@y@z$  where  $x,y,z \in \{0,1\}^*$  and  $x \geq 1$ . Give a Turing machine  $M_1$  to compute  $f_1$ .

**Solution 3.1:**

**Problem 3.2 (Bonus, 0 points):** Let  $f_2 : \{0,1,@\}^* \rightarrow \{0,1,@\}^*$  be the partial function given by  $f_2(x@y@z) = (x-1)@y@(z+y)$  where  $x,y,z \in \{0,1\}^*$  and  $x \geq 1$ . Give a Turing machine  $M_2$  to compute  $f_2$ . You may use the machines claimed in previous problems even if you have not constructed them.

**Solution 3.2:**

**Problem 3.3 (20 points):** Describe in English the ingredients of a Turing Machine to compute the partial function  $\times : \{0,1,@\}^* \rightarrow \{0,1\}^*$  given by  $\times(x@y) = xy$ . You may use the machines claimed in previous problems even if you have not constructed them.

**Solution 3.3:**

**Problem 3.4 (Bonus, 0 points):** Give a complete Turing Machine to compute the partial function  $\times$  described in Problem 3.3.

**Solution 3.4:**

**Problem 3.5 (Bonus, 0 points):** Give a Turing Machine to compute the partial function  $\times$  described in Problem 3.3 for which, on an input of length  $n$ , the number of transitions it makes before halting is  $O(n^{10})$ . (You'll need a strategy different from the one suggested by 3.1 and 3.2.)

**Solution 3.5:**

**Problem 4:**

**Problem 4.1 (15 points):** Prove that there exists a function  $f : \{0, 1\}^{1000} \rightarrow \{0, 1\}$  which is not computed by any Turing Machine with alphabet  $\{\triangleright, 0, 1, \phi\}$  and at most 10 states. (Hint: Give an upper bound on the number of Turing Machines with  $k$  states as a function of  $k$ .)

**Solution 4.1:**

**Problem 4.2 (5 points):** Prove that every function  $f : \{0, 1\}^{1000} \rightarrow \{0, 1\}$  is computed by a Turing Machine with alphabet  $\{\triangleright, 0, 1, \phi\}$ .

**Solution 4.2:**