Problem Set 3

Due: **9:59pm, Monday, 13 February**

This problem set focuses on understanding finite computation in the Boolean circuit and programming models (Chapter 3 of TCS), and understanding how we prove Boolean circuits can compute every finite function (Chapter 4 of TCS).

There are both overleaf and Jupyter notebook parts for this problem set. See https://uvatoc.github.io/ps/ps3.ipynb for the Jupyter part of this problem set.

You should complete the assignment by writing your answers in the ps3.tex LaTeX template and completing the Jupyter notebook, similarly to how you did Problem Set 1. You will submit your solutions to both parts separately in GradeScope, as you did for PS1.

Collaboration Policy: You may discuss the problems with anyone you want. You are permitted to use any resources you find for this assignment **other than solutions from previous cs3102/cs3120 courses**. You should write up your own solutions and understand everything in them, and submit only your own work. You should note in the *Collaborators and Resources* box below the people you collaborated with and any external resources you used (you do not need to list resources you used for help with LaTeX or Jupyter/Python).

Collaborators and Resources: TODO: replace this with your collaborators and resources (if you did not have any, replace this with *None*)

To do the LaTeX part of this assignment, follow the directions from Problem Set 1. The URL for the template is: https://www.overleaf.com/read/kvfjnywhnxrw

Before submitting your ps3.pdf file, also remember to:

– List your collaborators and resources, replacing the TODO in `\collaborators{TODO: replace ...}` with your collaborators and resources. (Remember to update this before submitting if you work with more people.)

– Replace the second line in ps3.tex, `\usepackage{uvatoc}` with `\usepackage[response]{uvatoc}` so the directions do not appear in your final PDF.

Problem 1 NOR *is universal (based on Exercise 3.7 in TCS book)*

Let *NOR* : $\{0,1\}^2 \to \{0,1\}$ defined as $NOR(a,b) = NOT(OR(a,b))$. Prove that $\{NOR\}$ is a universal set of gates. (Hint: show that anything that can be computed using *AND*,*OR*, and *NOT* can also be computed using just *NOR*).

Answer:

Problem 2 *XOR is not universal (based on Exercise 3.5 in TCS book)*

Prove that the set $XOR, 0, 1$ is not universal. (You can use any strategy you want to prove this; see the book for one hint of a possible strategy, but we think you may be able to find easier ways to prove this, and it is not necessary to follow the strategy given in the book.)

Answer:

Problem 3 *The MAJority is not Universal*

Prove that $\{\texttt{MAJ}, \texttt{0}, \texttt{1}\}$ is not a universal gate set (where `MAJ` is the majority of three inputs function (as defined in the Jupyter notebook), and `0` and `1` are constants).

Answer:

Problem 4 *Prove that a perceptron cannot compute* XOR.

A *perceptron* is a single layer neural network (Section 3.4.5) that can be modeled by the following function:

$$f(x_0, x_1, \ldots, x_{k-1}) = \sigma(\sum_i w_i x_i)$$

where $\sigma : \mathbb{R} \to \mathbb{R}$ is an activation function. For this question, you may assume the activation function is a rectified linear unit (ReLU), commonly used in deep learning:

$$ReLU(x) = \max(0, x)$$

Prove that there is no way to define *XOR* using a perceptron. That is, show that there is no way to assign the values of w_i such that $f(x_0, x_1) = ReLU(w_0 x_0 + w_1 x_1)$ implements the *XOR* function. You can interpret the output of f ias a Boolean value with values below 0.5 interpreted as False and values ≥ 0.5 interpreted as True (Exercise 3.11 uses a different interpretation which is more complex; if you prefer to use that one instead, this is fine.)

Historical and resource policy note: The proof that a perceptron cannot compute *XOR* is of some historical importance, and it doesn't take much cleverness to find proofs of this (don't click on this link until after submitting your assignment). You should not search for solutions to this problem since the goal of it is for you to think about this yourself and come up with a proof. We think it will be pretty obvious if you write-up a found proof, so expect everyone to be able to explain their proof and how they derived it to us orally if asked. The historical significance of this problem, which is often overblown to the point where some refer to it as the "XOR affair", is that it has been attributed by some as one of the reasons why research in neural networks mostly ceased in the 1980s, except for a few die-hard believers who kept working on it, eventually leading to the explosion of "deep learning" over the past decade, and being awarded the Turing Award in 2018.

Answer:

Problem 5 ($\star$) *Prove that a two-layer perceptron (using the same activation function as in the previous problem) is universal. You may assume that in addition to the perceptron the constant value 1 is available.*

This will require a bit of creativity and thinking carefully about our definitions. As in the previous problems, *universal* means that a Boolean circuit where the gates are all two-layer perceptrons can compute the same function as any Boolean circuit.

This is the end of the LaTeX problems for PS3. Remember to follow the last step in the directions on the first page to prepare your PDF for submission, and to also complete the problems in the Jupyter notebook, see https://uvatoc.github.io/ps3/.

Die weitere Erschließung dieses Feldes ist Aufgabe der Zukunft.

