Preparing for Exam 1

Exam 1 will be in class on Thursday, 2 March.

It will cover material from Classes 1–9, Problem Sets 1–4 (including the provided comments), Quizzes 1–4, and Chapter 1–5 of the TCS Book. Nearly everything on the exam will have been covered in at least three of these places (e.g., in class, on a problem set, and in the textbook; or in multiple classes, the textbook, and a quiz).

As a reminder from the syllabus, you may prepare a one-page (letter-size, two-sided) reference sheet for use during the exam, but all other resources are forbidden (no internet, textbook, other humans, magnification instruments, etc.). We expect that students will benefit from thinking about what to put on your reference sheet in preparing for the exam, and you may work with anyone you want (including other students in the class) to prepare your reference sheets together.

The problems below should give you an idea what to expect on the Exam — problems 1–9 are essentially an example of a full Exam 1 (from the Fall 2019 course, but with a few edits to some questions and whitespace removed), so similar in length to what you should expect for the exam on March 2. If you want to see the original exam that was given to students in 2019, you are welcome to look at that also: https://uvatoc.github.io/f19/exam1comments/. This exam does not cover some of the topics we have covered for this Exam, so we have also provided some additional practice problems after the practice exam.

We will post solutions and comments for some of these problems soon (and are happy to answer any questions you have about them), but encourage you to first try them on your own, then in discussion with other students if possible, before looking at the solutions (hopefully to verify your own answer is correct, and to see other approaches). We emphasize, though, that you are not expected to solve all of these problems and there is no submission expected for these. The problems are provided to give you problems to practice and check your understanding in preparing for the exam. If you are able to solve these problems, you should be confident that you'll be able to do well on the exam.

Practice Exam 1

Boolean Circuits

For these questions, we assume the following logical functions with their standard meanings:

> $\mathit{NOT}(a)$: $\mathit{NOT}(0) = 1, \mathit{NOT}(1) = 1$.
> $\mathit{OR}(a, b)$: $\mathit{OR}(0, 0) = 0$, otherwise $\mathit{OR}(a, b) = 1$.
> $\mathit{AND}(a, b)$: $\mathit{AND}(1, 1) = 1$, otherwise $\mathit{AND}(a, b) = 0$.

1. Give a simple description (which could be just the name of a well known function) of the function defined by the code below:

```
def MYSTERY(a, b):
   v1 = NAND(a, b)
   return NAND(v1, v1)
```

2. Define XOR using only NOT, OR and AND, where $\mathit{XOR}(0, 0) = \mathit{XOR}(1, 1) = 0$ and $\mathit{XOR}(0, 1) = \mathit{XOR}(1, 0) = 1$.

Countability

For these problems, you may use any results that were proven in class or on a problem set in your proof (without needing to prove them).

3. Prove that the set of all fish in the sea is *countable*. (For purposes of this question, you can assume the "sea" in question is the Mediterranean Sea, and *fish* has its conventional meaning.)

4. Prove that the set of the even natural numbers (i.e., $\{0, 2, 4, 6, \ldots\}$) is *countably infinite*.

5. Prove that the set of all directed graphs (as defined below) is *uncountable*.

> **Definition 1 (Directed Graph)** A *directed graph* $G = (V, E)$ consists of a (possibly infinite) set V of vertices and a (possibly infinite) set E of edges. Every edge is an ordered pair of two distinct elements of V.

Proofs with Definitions

Here we define the Counting Numbers, similarly to the definition of Natural Numbers you have seen in class:

> **Definition 2 (Counting Numbers)** We define the *Counting Numbers* as:
>
> 1. **1** is a Counting Number.
> 2. If n is a Counting Number, **S**(n) is a Counting Number.

6. Prove that the cardinality of the set of all *Counting Numbers* (as defined above) is *countably infinite*.

Computing Models

Recall from class and the book that two computing models are *equivalent* in power if every computation that can be defined using the first model can also be defined using the second model, and every computation that can be defined using the second model can also be defined using the first model.

7. Prove that Boolean circuits using the gateset $\{MAJ, NOT, ZERO\}$ is equivalent to Boolean circuits using the gateset $\{MAJ, NOT, ONE\}$. ($ZERO$ is the constant gate that outputs `0` for any input, and ONE is the constant gate that outputs `1` for any input. MAJ is the majority of three inputs gate you are familiar with from PS3.)

8. Prove that Boolean circuits using the gateset $\{MAJ, NOT\}$ is **not** equivalent to Boolean circuits using the gateset $\{NAND, XOR\}$.

9. Prove that *AON-STRAIGHTLINE*, straightline programs composed of *AND*, *OR*, and *NOT* operations is equivalent to *MOP-STRAIGHTLINE*, straightline programs composed of the plus, multiply, and constant one operations defined by:

```
def PLUS(a, b):
   return (a + b) % 2


def MULT(a, b):
   return (a * b) % 2


def ONE(a, b):
   return 1
```

The `%` operator is modulo (remainder after division). So, for example `(0 + 1) % 2 = 1`, `(1 + 1) % 2 = 0`, and `(1 * 1) % 2 = 1`.

End of Practice Exam 1

Additional Practice Problems

Below are some additional practice problems that we think will be helpful to you in preparing for the Exam. They are collected and adapted from various sources, mostly previous exams used in our courses, so should give you a good idea of other types of problems to expect on the exam.

Countable, Uncountable, Unknown

10. For each set described below, indicate whether its cardinality is *Countable*, *Uncountable*, or *Unknown* (not determined by the question if it is countable or uncountable). Circle one option and give a proof of your answer.

(a) The set of all grades that students will get on this exam.

Countable *Uncountable* *Unknown*

Proof:

(b) The set of NAND circuits that compute XOR.

Countable *Uncountable* *Unknown*

Proof:

(c) The set of of all Boolean functions that cannot be computed using a Boolean circuit with 2^{3120} or fewer NAND gates.

Countable *Uncountable* *Unknown*

Proof:

Induction

11. Define the function $\texttt{ALT}_n : \{0,1\}^{2n} \to \{0,1\}$ such that for a string $w \in \{0,1\}^{2n}$ we say that $\texttt{ALT}_n(w) = 1$ provided $w \in (01)^*$. We could compute $\texttt{ALT}_1$ using the following straightline program:

```
def ALT1(x1,x2):
    diff = XOR(x1,x2)
    return AND(x2, diff)
```

We could then implement $\texttt{ALT}_n$ as follows:

```
def ALTn(x1,x2,...,x2n):
    diff = XOR(x1,x2)
    first = AND(x2,diff)
    rest = ALTn(x3,...,x2n)
    return AND(first, rest)
```

Suppose we have similarly implemented $\texttt{ALT}_{n-1}$, $\texttt{ALT}_{n-2}$, etc., (and all other dependent subroutines).

Show that the number of `NAND` gates needed to represent a circuit for $\texttt{ALT}_n$ is no more than $10n$ gates (hint: `XOR` requires 4 `NAND` gates and `AND` requires 3 `NAND` gates).

Asymptotics

12. Let $f(n) = 8n^{4.5}$ and $g(n) = 5n^5$, which of the following are true? Support your answer to each part with a convincing argument.

(a) $f \in O(g)$

(b) $f \in \Omega(g)$

(c) $f \in \Theta(g)$

Relation Properties

13. Considered the relation, $\leq$ (less than or equal to, with the standard meaning), with the domain set, $\mathbb{N}$ and codomain set $\mathbb{N}$. Which of these properties does the $\leq$ relation have: function, total, injective, surjective, bijective?

14. Set Cardinality

a. Assume $R : A \to B$ is an *total injective* function between A and B. What must be true about the relationship between $|A|$ and $|B|$?

b. Assume $R : A \to B$ is an *total surjective* function between A and B. What must be true about the relationship between $|A|$ and $|B|$?

c. Assume $R : A \to B$ is a (not necessarily total) *surjective* function between A and B. What must be true about the relationship between $|A|$ and $|B|$?

Countable and Uncountable Infinities

15. Prove that the integers, i.e., $\ldots, -2, -1, 0, 1, 2, \ldots$, are countably infinite.

16. Prove that the number of total injective functions between $\mathbb{N}$ and $\mathbb{N}$ is countable.

17. Prove that the number of different chess positions is countable. (A chess position is defined by the locations of pieces on an 8×8 board, where each square on the board can be either empty, or contain a piece from $\{\text{Pawn}, \text{Knight}, \text{Bishop}, \text{Castle}, \text{Queen}, \text{King}\}$ of one of two possible colors.)

18. Prove that number of Ziggy Pig ice cream dishes is uncountable. A Ziggy Pig ice cream can contain any number of scoops ($scoops \in \mathbb{N}$), and each scoop can be of any flavor, where distinct flavors are identified by $v \in \mathbb{N}$.

19. Proof that all fish who have fully eaten Ziggy Pig ice creams (as describe in the previous problem) with an infinite number of scoops are Coho Salmon.

Induction Practice

20. Prove by induction that every natural number less than 2^{k+1} can be written as $a_0 \cdot 2^0 + a_1 \cdot 2^1 + a_2 \cdot 2^2 + \cdots + a_k \cdot 2^k$ where all the a_i values are either 0 or 1.

21. Prove by induction that every finite non-empty subset of the natural numbers contains a *greatest* element, where an element $x \in S$ is defined as the *greatest* element if $\forall z \in S - \{x\}.\ x > z$.

22. In class, we argued that a "good" Boolean circuit always eventually evaluates to a value using the definition of circuit evaluation. Prove that a Boolean circuit where there is a cycle on a path between an input and an output will never produce a value for that output.

