CS 70 Summer 2018 — Discrete Mathematics and Probability Theory — Sinho Chewi and Vrettos Moulos — HW 4

Due: Friday, 7/13, 10 PM

# Sundry

Before you start your homework, write down your team. Who else did you work with on this homework? List names and email addresses. (In case of homework party, you can also just describe the group.) How did you work on this homework? Working in groups of 3-5 will earn credit for your "Sundry" grade.

Please copy the following statement and sign next to it:

*I certify that all solutions are entirely in my words and that I have not looked at another student's solutions. I have credited all external sources in this write up.*

# 1 Countability Introduction

(a) Do $(0,1)$ and $\mathbb{R}_+ = (0,\infty)$ have the same cardinality? If so, give an explicit bijection (and prove that it's a bijection). If not, then prove that they have different cardinalities.

(b) Is the set of English strings countable? (Note that the strings may be arbitrarily long, but each string has finite length.) If so, then provide a method for enumerating the strings. If not, then use a diagonalization argument to show that the set is uncountable.

(c) Consider the previous part, except now the strings are drawn from a countably infinite alphabet $\mathscr{A}$. Does your answer from before change? Make sure to justify your answer.

# 2 Counting Cartesian Products

For two sets $A$ and $B$, define the Cartesian Product as $A \times B = \{(a,b) : a \in A, b \in B\}$.

(a) Given two countable sets $A$ and $B$, prove that $A \times B$ is countable.

(b) Given a finite number of countable sets $A_1, A_2, \dots, A_n$, prove that $A_1 \times A_2 \times \cdots \times A_n$ is countable.

(c) Consider an infinite number of countable sets: $B_1, B_2, \dots$. Under what condition(s) is $B_1 \times B_2 \times \cdots$ countable? Prove that if this condition is violated, $B_1 \times B_2 \times \cdots$ is uncountable.

# 3 Impossible Programs

Show that none of the following programs can exist.

(a) Consider a program $P$ that takes in any program $F$, input $x$ and output $y$ and returns true if $F(x)$ outputs $y$ and returns false otherwise.

(b) Consider a program $P$ that takes in any program $F$ and returns true if $F(F)$ halts and returns false if it doesn't halt.

(c) Consider a program $P$ that takes in any programs $F$ and $G$ and returns true if $F$ and $G$ halt on all the same inputs and returns false otherwise.

# 4 Printing All $x$ where $M(x)$ Halts

(a) Prove that it is possible to write a program $P$ which:

- takes as input $M$, a Java program,
- runs forever, and prints out strings to the console,
- for every $x$, if $M(x)$ halts, then $P(M)$ eventually prints out $x$,
- for every $x$, if $M(x)$ does NOT halt, then $P(M)$ never prints out $x$.

(b) Lexicographical ordering of strings means (1) shorter strings are in front of longer strings and (2) for two strings of the same length, they are sorted in alphabetical order.

Prove that it's impossible to solve the above problem if we require the output be in lexicographical order.

# 5 Counting, Counting, and More Counting

The only way to learn counting is to practice, practice, practice, so here is your chance to do so. For this problem, you do not need to show work that justifies your answers. We encourage you to leave your answer as an expression (rather than trying to evaluate it to get a specific number).

(a) How many ways are there to arrange $n$ 1s and $k$ 0s into a sequence?

(b) A bridge hand is obtained by selecting 13 cards from a standard 52-card deck. The order of the cards in a bridge hand is irrelevant.
How many different 13-card bridge hands are there? How many different 13-card bridge hands

are there that contain no aces? How many different 13-card bridge hands are there that contain all four aces? How many different 13-card bridge hands are there that contain exactly 6 spades?

(c) Two identical decks of 52 cards are mixed together, yielding a stack of 104 cards. How many different ways are there to order this stack of 104 cards?

(d) How many 99-bit strings are there that contain more ones than zeros?

(e) An anagram of FLORIDA is any re-ordering of the letters of FLORIDA, i.e., any string made up of the letters F, L, O, R, I, D, and A, in any order. The anagram does not have to be an English word.
How many different anagrams of FLORIDA are there? How many different anagrams of ALASKA are there? How many different anagrams of ALABAMA are there? How many different anagrams of MONTANA are there?

(f) How many different anagrams of ABCDEF are there if: (1) C is the left neighbor of E; (2) C is on the left of E.

(g) We have 9 balls, numbered 1 through 9, and 27 bins. How many different ways are there to distribute these 9 balls among the 27 bins? Assume the bins are distinguishable (e.g., numbered 1 through 27).

(h) We throw 9 identical balls into 7 bins. How many different ways are there to distribute these 9 balls among the 7 bins such that no bin is empty? Assume the bins are distinguishable (e.g., numbered 1 through 7).

(i) How many different ways are there to throw 9 identical balls into 27 bins? Assume the bins are distinguishable (e.g., numbered 1 through 27).

(j) There are exactly 20 students currently enrolled in a class. How many different ways are there to pair up the 20 students, so that each student is paired with one other student?

(k) How many solutions does $x_0 + x_1 + \cdots + x_k = n$ have, if each $x$ must be a non-negative integer?

(l) How many solutions does $x_0 + x_1 = n$ have, if each $x$ must be a *strictly positive* integer?

(m) How many solutions does $x_0 + x_1 + \cdots + x_k = n$ have, if each $x$ must be a *strictly positive* integer?

# 6 Fermat's Wristband

Let $p$ be a prime number and let $k$ be a positive integer. We have beads of $k$ different colors, where any two beads of the same color are indistinguishable.

(a) We place $p$ beads onto a string. How many different ways are there construct such a sequence of $p$ beads with up to $k$ different colors?

(b) How many sequences of $p$ beads on the string are there that use at least two colors?

(c) Now we tie the two ends of the string together, forming a wristband. Two wristbands are equivalent if the sequence of colors on one can be obtained by rotating the beads on the other. (For instance, if we have $k = 3$ colors, red (R), green (G), and blue (B), then the length $p = 5$ necklaces RGGBG, GGBGR, GBGRG, BGRGG, and GRGGB are all equivalent, because these are all rotated versions of each other.)

    How many non-equivalent wristbands are there now? Again, the $p$ beads must not all have the same color. (Your answer should be a simple function of $k$ and $p$.)

    [*Hint*: Think about the fact that rotating all the beads on the wristband to another position produces an identical wristband.]

(d) Use your answer to part (c) to prove Fermat's little theorem.