CS 70 Fall 2025 — Discrete Mathematics and Probability Theory — Sabin, Hug — HW 08

Due: Saturday, 10/25, 4:00 PM
Grace period until Saturday, 10/25, 6:00 PM
Remember to show your work for all problems!

Sundry

Before you start writing your final homework submission, state briefly how you worked on it. Who else did you work with? List names and email addresses. (In case of homework party, you can just describe the group.) If you used an LLM, place transcripts of your chats here.

1 Post-Midterm Check-In Form

Please fill out this required form to let us know how the midterm went for you and what feedback you have for us: https://forms.gle/V7JgZnExmFqCAwjR7!

2 Count It!

Note 11 For each of the following collections, determine and briefly explain whether it is finite, countably infinite (like the natural numbers), or uncountably infinite (like the reals):

(a) The integers which divide 8.

(b) The integers which 8 divides.

(c) The functions from $\mathbb{N}$ to $\mathbb{N}$.

(d) The set of strings over the English alphabet. (Note that the strings may be arbitrarily long, but each string has finite length. Also the strings need not be real English words.)

(e) The set of finite-length strings drawn from a countably infinite alphabet, $\mathscr{C}$.

(f) The set of infinite-length strings over the English alphabet.

3 Fixed Points

Note 12 Consider the problem of determining if a program P has any fixed points. Given any program P, a fixed point is an input x such that $P(x)$ outputs x.

(a) Prove that the problem of determining whether a program has a fixed point is uncomputable.

(b) Consider the problem of outputting a fixed point of a program if it has one, and outputting "Null" otherwise. Prove that this problem is uncomputable.

(c) Consider the problem of outputting a fixed point of a program F if the fixed point exists *and* is a natural number, and outputting "Null" otherwise. If an input is a natural number, then it has no leading zero before its most significant bit.

Show that if this problem can be solved, then the problem in part (b) can be solved. What does this say about the computability of this problem? (You may assume that the set of all possible inputs to a program is countable, as is the case on your computer.)

4 Five Up

Note 13

Say you toss a coin five times, and record the outcomes. For the three questions below, you can assume that order matters in the outcome, and that the probability of heads is some p in $0 < p < 1$, but *not* that the coin is fair ($p = 0.5$).

(a) What is the size of the sample space, $|\Omega|$?

(b) How many elements of Ω have exactly three heads?

(c) How many elements of Ω have three or more heads?

For the next three questions, you can assume that the coin is fair (i.e. heads comes up with $p = 0.5$, and tails otherwise).

(d) What is the probability that you will observe the sequence HHHTT? What about HHHHT?

(e) What is the probability of observing at least one head?

(f) What is the probability you will observe more heads than tails?

5 Aces

Note 13

Consider a standard 52-card deck of cards, which has 4 suits (hearts, diamonds, clubs, and spades) with 13 cards in each suit. Each suit has one ace. Hearts and diamonds are red, while clubs and spades are black.

(a) Find the probability of getting an ace or a red card, when drawing a single card.

(b) Find the probability of getting an ace or a spade, but not both, when drawing a single card.

(c) Find the probability of getting the ace of diamonds when drawing a 5 card hand.

(d) Find the probability of getting exactly 2 aces when drawing a 5 card hand.

(e) Find the probability of getting at least 1 ace when drawing a 5 card hand.

(f) Find the probability of getting at least 1 ace or at least 1 heart when drawing a 5 card hand.

6 Past Probabilified

Note 13 In this question we review some of the past CS70 topics, and look at them probabilistically.

For the following experiments, define an appropriate sample space Ω, and give the probability function $\mathbb{P}[\omega]$ for each $\omega \in \Omega$. Then compute the probabilities of the events E_1 and E_2.

(a) Fix a prime $p > 2$, and uniformly sample twice with replacement from $\{0, \ldots, p-1\}$ (assume we have two $\{0, \ldots, p-1\}$-sided fair dice and we roll them). Then multiply these two numbers with each other in $\pmod p$ space.

$$E_1 = \text{The resulting product is } 0.$$
$$E_2 = \text{The product is } (p-1)/2.$$

(b) Make a graph on n vertices by sampling uniformly at random from all possible edges, (assume for each edge we flip a coin and if it is head we include the edge in the graph and otherwise we exclude that edge from the graph).

$$E_1 = \text{The graph is complete.}$$
$$E_2 = \text{vertex } v_1 \text{ has degree } d.$$

(c) Create a random stable matching instance by having each person's preference list be a random permutation of the opposite entities (make the preference list for each individual job and each individual candidate a random permutation of the opposite entities). Finally, create a uniformly random pairing by matching jobs and candidates up uniformly at random (note that in this pairing, (1) a candidate cannot be matched with two different jobs, and a job cannot be matched with two different candidates (2) the pairing does not have to be stable).

$$E_1 = \text{All jobs have distinct favorite candidates.}$$
$$E_2 = \text{The resulting pairing is the candidate optimal stable pairing.}$$