CS 70 Discrete Mathematics and Probability Theory
Summer 2020 Amin Ghafari, Yining Liu, Khalil Sarwari

Midterm

- You may consult two handwritten double-sided sheets of notes. Apart from that, you may not look at books, notes, etc. Calculators, phones, computers, and other electronic devices are prohibited unless they are part of the recording submission. The only tabs/windows you may have open are the Exam PDF, Exam Google Doc, Midterm Instructions Doc and/or Exam Policy (Public) Doc, timer/clock, and Zoom.
- There are 9 questions on this exam, worth a total of 100 points.
- We will not take any clarifying questions; if there is a mistake in the exam, we will resolve it via regrade request or remove the corresponding question/part entirely.
- The questions vary in difficulty, so if you get stuck on any question it may help to leave it and return to it later.
- **Make sure you read the title of the first problem out loud**
- **You may, without proof, use theorems and facts that were proven in the lecture, notes, discussions, and/or in homeworks unless explicitly mentioned otherwise.**
- **You have 120 minutes to work on the exam. After this time, you may no longer work on the exam. You will then have 30 minutes for scanning and uploading your answers. Late submissions will be penalized.**
- **When uploading the exam PDF to Gradescope, you must state out loud *"I, [name], finished the exam entirely on my own, and submitted the exam to Gradescope at [time]."***

1 (10 points) Logic Questions

Before starting this question, please read the title out loud: "Logic Questions".
This is to verify it is not a prerecorded video. Not doing this will void your exam.

(a) (3 points) **First** write each of the following two statements in the form "$p \implies q$" in English. **Then** state either the converse or contrapositive of each of the statements in English as well.
For example: for the statement "I get up early on weekdays", the statement in English is "weekdays $\implies$ get up early". For each of your six answers, you may use the implication symbol $\implies$, but should otherwise use only English:

(i) I always listen to music when I am happy.

Statement:

Contrapositive:

(ii) I will answer your question only if you make a Piazza post.

Statement:

Converse:

(b) (2 points) Let $P(x)$ be the statement $x = x^2$. If the domain consists of integers, what are the truth values of the proposition $P(0)$ and the proposition $\forall x\, P(x)$?

(c) (2 points) Let $B(x)$ be the statement "x likes boba" and $J(x)$ be the statement "x can program in Java" and $F(x,y)$ be the statement "x and y are friends". The domain of quantifiers consists of all students in CS70 Summer 2020. Express the following sentence in terms of $B(x)$, $J(x), F(x,y)$, quantifiers, and logical connectives.

"Every student has a friend in CS70 who likes boba but doesn't know how to program in Java."

(d) (3 points) Prove $(\neg Q \wedge (P \implies Q)) \implies \neg P$ is a tautology using logical equivalences (i.e. answers that use truth tables or explaining in words will get zero points).

2 (16 points) Proofs

(a) (3 points) The following is a false proof. **Locate** (state which step is wrong) and **explain** the error.

Proposition: Given $n \in \mathbb{N}$, for any set S with $|S| = n$, then S must be a set of n zeros.

Proof:

Step 1: If $n = 0$, S is the empty set, thus the statement is trivially true.

Step 2: Let $k \in \mathbb{N}$. Assume the statement is true for a set S whenever $|S| \leq k$.

Step 3: Now consider a set S where $|S| = k+1$.

Step 4: Divide $S = S_1 \cup S_2$ such that $S_1 \cap S_2 = \emptyset$ and $0 < |S_1|, |S_2| < k+1$.

Step 5: Apply the inductive hypothesis on S_1, S_2, we see that S_1, S_2 contain only zeros. Since $S = S_1 \cup S_2$, we know S also only contain zeros.

(b) (5 points) Fibonacci sequence is defined such that each number is the sum of the two preceding ones, starting from 0 and 1. That is

$$F_0 = 0, \quad F_1 = 1, \qquad F_n = F_{n-1} + F_{n-2}.$$

Prove by induction that $F_n^2 - F_{n+1}F_{n-1} = (-1)^{n-1}$.

(HINT: You may need to use the property $F_n = F_{n-1} + F_{n-2}$ a couple of times in your inductive step of the proof. Your proof shouldn't be very long!)

(c) (5 points) Let $f : X \to Y$. Recall that for any $S \subseteq X$, $f(S) = \{f(s) : s \in S\}$.

(i) Given $A, B \subseteq X$, prove that $f(A) \setminus f(B) \subseteq f(A \setminus B)$.

(ii) Give an example of A, B and $f : X \to Y$ such that $f(A) \setminus f(B) \neq f(A \setminus B)$

(d) (3 points) Prove that there are no solutions in integers x and y to the equation $x^2 - 3y^3 = 2$.
(HINT: Consider the equation modulo 3.)

3 (8 points) Tiny Sets

Given $X \subseteq \mathbb{R}$, we say X is *tiny* if for each $\varepsilon > 0$, we can find a countable collection of closed intervals $[a_1, b_1], [a_2, b_2], \ldots$ such that

(i) $X \subseteq \bigcup_{n=1}^{\infty} [a_n, b_n]$;

(ii) $\sum_{n=1}^{\infty} (b_n - a_n) < \varepsilon$.

Intuitively, X is tiny if we can find an arbitrarily small cover of X using closed intervals.

(a) (2 points) Prove that any singleton set, i.e. $\{x\}$ where $x \in \mathbb{R}$, is tiny.

(b) (i) (4 points) Let $A_1, A_2, \ldots$ be a countable collection of tiny sets. Prove $\bigcup_{n=1}^{\infty} A_n$ is tiny.
(HINT: you may find $\sum_{n=1}^{\infty} \frac{1}{2^n} = 1$ *to be helpful.)*

(ii) (2 points) Is $\mathbb{Q}$ tiny? Justify your answer.

4 (8 points) Computability

Let P and Q be programs (finite length bit-strings), and x is an input (finite length bit-string). Consider the program $FASTER(P,Q,x)$ which returns True if $P(x)$ takes strictly fewer steps than $Q(x)$ to execute, and returns False otherwise. Note: If $P(x)$ and $Q(x)$ both take infinitely many steps, then $FASTER$ returns False.

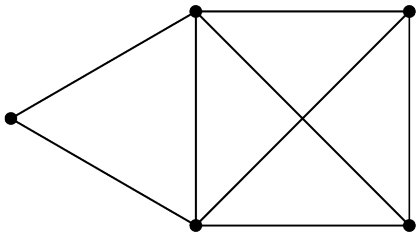
(a) (3 points) Is $FASTER$ computable? If so, provide pseudocode for the program $FASTER$. If not, prove that it is uncomputable.

(b) (5 points) Construct a program $ALLFASTER(x)$ which prints out all tuples (P,Q) satisfying the condition where $P(x)$ takes strictly fewer steps than $Q(x)$ to execute. For every pair of programs, P,Q where $P(x)$ terminates before $Q(x)$ the tuple (P,Q) must be printed out after a finite number of steps. Tuples may be printed out multiple times so long as they satisfy the condition.

5 (12 points) Chromatic Numbers

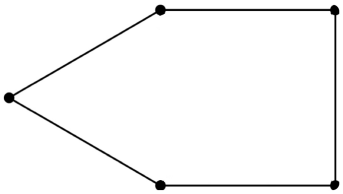
The *chromatic number* of a graph $G = (V, E)$, written $\chi(G)$, is the minimum number of colors required to assigned a color to each vertex of G such that no two adjacent vertices are assigned the same color.

Now, define the *clique number* of a graph $G = (V, E)$, written $\omega(G)$, to be the number of vertices in the largest complete subgraph of G.

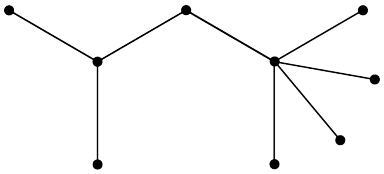


$\chi(G) = 4$ and $\omega(G) = 4$ for the graph G above.

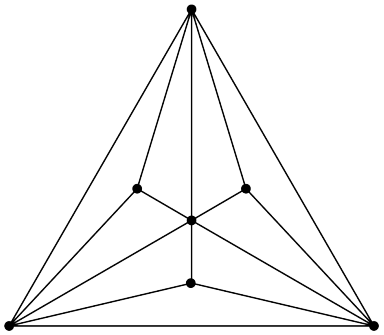
(a) (3 points) Determine the chromatic number and the clique number of each of the following graphs.



(i) $\chi(G_1) =$ $\omega(G_1) =$



(ii) $\chi(G_2) =$ $\omega(G_2) =$



(iii) $\chi(G_3) =$ $\omega(G_3) =$

(b) (5 points) Let K_n be the complete graph on n vertices. Determine an expression for $\chi(K_n)$ in terms of only n as a variable. Prove your result.

(c) (4 points) Prove that for any graph G, $\chi(G) \geq \omega(G)$.

6 (17 points) Cicada Party

There are two types of cicadas: type A comes out every 13 years (so A comes out on year $0, 13, 26, \ldots$), and type B comes out every 17 years, (so B comes out on year $0, 17, 34, \ldots$). Assume that it is currently year 0, and both types of cicadas come out this year. You will have to do some calculation in this question.

It may be useful to know: $13^{-1} \equiv 4 \pmod{17}$ and $17^{-1} \equiv 10 \pmod{13}$.

(a) (3 points) What is the next year that both types of cicadas come out?

(b) (5 points) A "so close" year is a year when one type of cicada comes out, but the other type of cicada came out the year before! For example, year 170 will be a "so close" year because B comes out in year 170 but A had already came out in 169.

What is the first "so close" year?

(c) (4 points) The cicadas celebrate a "lucky" year when A is 3 years in its cycle (e.g. year $3, 16 \ldots$) and B is 5 years in its cycle (e.g. year $5, 22, \ldots$).

What is the first "lucky" year?

(d) (5 points) It is now year 2020. How many "lucky" years have occurred since year 0?

7 (12 points) Key Exchange

Given a prime p, we say an integer $1 \leq g \leq p-1$ is a *primitive root* if for any $a \in \{1,\ldots,p-1\}$, there exists a unique $k \in \{1,\ldots,p-1\}$ such that $g^k \equiv a \pmod{p}$. In other words, exponentiation of g would give us all the elements in $\{1,\ldots,p-1\}$. It can be proven that a primitive root exists given any prime p.

(a) (1 point) Is the following statement correct? "1 is not a primitive root for all prime $p > 2$."

Briefly justify your answer.

Alice and Bob want to agree upon a key that they will use for future communications. They decide to use the following scheme:

Setup: They pick a prime p, a primitive root g and make (p,g) public. Alice and Bob each picks an integer between 1 and $p-1$ as their own private keys, i.e. Alice picks $s_A \in \{1,\ldots,p-1\}$ and Bobs picks $s_B \in \{1,\ldots,p-1\}$ without sharing the private keys with each other.

Protocol:

1. First, Alice computes $m_A = g^{s_A} \mod p$ and sends it to Bob, and Bob computes $m_B = g^{s_B} \mod p$ and sends it to Alice. Eve **can** see m_A and m_B.
2. Next, Alice computes $k_A = m_B^{s_A} \mod p$ and Bob computes $k_B = m_A^{s_B} \mod p$.

Note that Eve **cannot** see k_A or k_B since those are never shared over any network.

(b) (2 points) Recall that the goal is to agree upon a key for future communications. Show that $k_A = k_B$.

(c) (3 points) The *discrete logarithm problem* asks: given a prime p, a primitive root g, and an integer $1 \leq a \leq p-1$, find the $k \in \{1,\ldots,p-1\}$ such that $g^k \equiv a \pmod{p}$.

Show that if Eve can efficiently solve the discrete logarithm problem, she will be able to efficiently compute k_A.

(d) (4 points) Show that if Eve can solve the discrete logarithm problem efficiently, then she can also break RSA efficiently.

(HINT: Recall that breaking RSA means figuring out the private key d given only the public key (N,e), where N is the product of two large primes.)

(e) (2 points) Evil Eve was able to bribe Bob to give her s_B. Given knowledge of s_B, can Eve decrypt the messages that Alice and Bob are sharing? Keep in mind that Eve can see all the encrypted communications between Alice and Bob. (Eve cannot take the discrete logarithm!)

8 (12 points) GCD for Polynomials

A common divisor of $P(x)$ and $Q(x)$ is a polynomial $D(x)$ that divides $P(x)$ and $Q(x)$, i.e. $P(x) = D(x)P'(x)$ and $Q(x) = D(x)Q'(x)$ for some polynomials $P'(x)$ and $Q'(x)$. Furthermore, $D(x)$ is the greatest common divisor (GCD) of $P(x)$ and $Q(x)$, if every common divisor of $P(x)$ and $Q(x)$ also divides $D(x)$.

We can use the Euclidean algorithm to find the GCD of any pair of polynomials. It makes repeated use of Euclidean division. When using this algorithm on two numbers, the size of the numbers decreases at each stage. With polynomials, the degree of the polynomials decreases at each stage. The last nonzero remainder, made monic (i.e. it has 1 as coefficient of the highest degree) if necessary, is the GCD of the two polynomials.

Assume we wish to find $\mathrm{GCD}(P(x), Q(x))$ where we know:

$$\deg(Q(x)) \leq \deg(P(x))$$

We can find polynomials $A(x)$ and $B(x)$ such that

$$P(x) = A(x)Q(x) + B(x), \qquad \deg(B(x)) < \deg(Q(x)).$$

Thus, for the $\mathrm{GCD}(P_0 = P(x), Q_0 = Q(x))$ we have

$$\begin{aligned} D(x) &= \mathrm{GCD}(P_0 = P(x), Q_0 = Q(x)) \\ &= \mathrm{GCD}\Big(Q(x), P(x) \pmod{Q(x)}\Big) \\ &= \mathrm{GCD}\big(Q(x), B(x)\big) = \mathrm{GCD}(P_1, Q_1) \end{aligned}$$

We can repeat this for i steps:

$$D(x) = \mathrm{GCD}(P_0, Q_0) = \mathrm{GCD}(P_1, Q_1) = \cdots = \mathrm{GCD}(P_i, Q_i)$$

until $Q_{i=N}(x) = 0$, and the GCD is

$$D(x) = \mathrm{GCD}(P_0, Q_0) = \mathrm{GCD}(P_1, Q_1) = \cdots = \mathrm{GCD}(P_N, 0) = P_N$$

(a) (3 points) Use this algorithm to find the GCD of $P(x) = x^3 + x^2 + x + 1$ and $Q(x) = x^2 + x$.

(b) (4 points) We say two polynomials are *coprime* if they share no common polynomial factor, i.e. $P(x)$ being $Q(x)$ coprime means

$$\text{GCD}\big(P(x), Q(x)\big) = 1.$$

Show that if $P(x)$ and $Q(x)$ are coprime then the multiplicative inverse of $P(x)$ mod $Q(x)$ exists.

(HINT: Bezout's theorem for polynomials states that if $GCD\big(P(x), Q(x)\big) = D(x)$*, then there exist polynomials* $A(x)$ *and* $B(x)$ *such that* $D(x) = A(x)P(x) + B(x)Q(x)$*.)*

(c) (5 points) Show that $P(x) = x^3 + x^2 + 1$ and $Q(x) = x^2 + 1$ are coprime. Then, find the multiplicative inverse of $P(x)$ mod $Q(x)$.

9 (5 points) Degrading Channels

Alice would like to send a secure message to Bob over a channel of size n. This means Alice can send at most n packets at a time across the channel. However, the channel is not very reliable.

Assume the channel behaves as follows: Of the first batch of n packets, it corrupts none; of the second batch of n packets, it corrupts exactly 1; of the third batch of n, it corrupts exactly 2; and so on, until for the $(n+1)^{th}$ batch of n packets (and thereafter), it corrupts all of them.

Suppose we use error correcting codes for each batch of packets in order to recover the original messages which is sent through the channel. What is the maximum size message (in terms of packets) that we can send? **Justify your answer. Assume n is even.**

Your final answer should be a closed-form expression (not a summation).

Submission

- Keep the recording going;
- Scan answer booklet and cheatsheets into PDF;
- Submit to Gradescope by 10:30PM PDT. If Gradescope is being really slow, you may submit your exam PDF using this form: https://forms.gle/r79Ldpn9sBQsDzty (emergency only);
- State out loud *"I, [name], finished the exam entirely on my own, and submitted the exam to Gradescope at [time]."*;
- Stop the recording;
- Upload recording to your Google Drive;
- Submit Google Drive link to your uploaded recording using this form: https://forms.gle/3UEFBLgSuVbt6UKA6 by 11:59PM 7/14 PDT.
- Submit your cheatsheets to Gradescope under "Midterm Cheat Sheets". If you did not use any cheatsheets, you must submit 4 blank pages.

If you have technical issues during the exam, you should report these issues when you submit your exam by emailing su20@eecs70.org.