Midterm Exam

Question	Parts	Points
1: Read and Follow These Instructions	1	0
2: Where Did the Magic Go?	3	20
3: Pherekydes Triplets	1	15
4: Lights Out!	2	15
5: Just-in-Time Water Supply	2	30
6: Finding Duplicates	3	25
7: Car Dealer	2	20
8: The Angels are Playing	3	35
Total:		160

Name: ______________________________

Problem 1. [0 points] **Read and Follow These Instructions** (1 part)

- This open book, take-home mid-term exam is due on **Tuesday, April 7 at noon (ET)**. There won't be any allowances for late submissions so please plan ahead!
- Please refrain from discussing the exam with your class-mates until **Wednesday April 8**.
- While you are welcome to print the exam to work on it, all submissions must be made via Gradescope using a combination of Markup, KaTeX, and uploaded photos. To avoid any last-minute problems, we encourage you to start placing your responses into Gradescope early, and update them as you improve your solutions.
- If you have any technological difficulties relating to this exam, please email *karchmer@mit.edu*.
- This is an individual exam. *No collaboration is allowed.*
- You can use any of the official course materials from this semester's edition of the class, including lecture notes and recordings, recitation notes, problem sets and solutions, handouts and the textbook (CLRS).
- You *cannot use* any other materials or resources that were not explicitly allowed in the previous point, including material from previous editions of this course, any other textbook and/or online resource.
- You cannot use the online resources, except the ones described above and to get help regarding LaTeX and Gradescope.
- You can ask questions about the exam in Piazza via private posts. *Posting public posts is not allowed.*
- You have time to write well-written answers that are clear and touch on every important aspect of the correct solution. Full credit will only be given for answers that are crafted accordingly.

Problem 2. [20 points] **Where Did the Magic Go?** (3 parts)
Alyssa P. Hacker has a graph with n vertices $V = \{v_1, \ldots, v_n\}$ such that one of the graph edges is "magic." Your task is to devise an algorithm that will identify this magic edge by iteratively querying Alyssa. Each query presents her with a subset A of vertices V to which she will respond with one of the following answers:

- YES, indicating that the magic edge is between a vertex in A and a one in $V \setminus A$; and
- NO, otherwise.

(a) [6 points] Assume that you already found a set A such that Alyssa responds YES to the corresponding query. (We will call such set *bridging*.) Give an algorithm that identifies the magic edge using $O(\log n)$ further queries.

(b) [7 points] Design a *randomized* algorithm that finds a set A that is bridging (as described in Part (a)) using $O(1)$ queries in expectation.

(c) [7 points] Give a *deterministic* algorithm that finds a bridging set using at most $O(\log n)$ queries.

Problem 3. [15 points] **Pherekydes Triplets** (1 part)
Pherekydes, Pythagoras' teacher, is jealous. Those triplets his pupil defined are really great. Not to be outdone, he makes up his own definition. A Pherekydes triple (pheryple in short) is a triple of numbers $a < b < c$ that satisfies the equation $b - a = c - b$.

Given a set A with n distinct integers $a_1, a_2, \ldots, a_n$ in the range $[0, m]$ find the number of pheryples with numbers from the set A. Credit will only be given for algorithms that are $o((n + m)^2)$.

For full credit, you should: (i) give an algorithm, (ii) justify its correctness and (iii) state and briefly justify its running time.

Problem 4. [15 points] **Lights Out!** (2 parts)
Oh no! The power went out just as students are arriving in 34-101 for their favorite lecture. Students don't have a choice but to enter the darkened room one-at-a-time and try to sit in a random seat. If that seat is occupied, they try to sit in a second random seat (which can be the same seat again). If that seat is also occupied, our helpful TAs come to the rescue and guide the student to an unoccupied seat (after the second unsuccessful attempt) before calling in the next student. (That is, each student is seated after at most two attempts, and the seating occurs sequentially.)

(a) [8 points] Show that the expected number of students that require help is at most $\frac{n^3}{3m^2}$, where n is the number of students and m is the number of seats in 34-101.
Hint: You may find the inequality $\sum_{i=0}^{n-1} i^2 \leq \frac{n^3}{3}$ *helpful.*

(b) [7 points] The Registrar assures us that we have twice as many seats as students. Show that the probability that more than a third of the students will need help is at most $e^{-\frac{1}{12}n}$. You can assume the result of Part (a).

Problem 5. [30 points] **Just-in-Time Water Supply** (2 parts)
Congratulations! Your MISTI-Greece internship came through. You are spending IAP on a sunny Greek island, managing the sea water supply for the bath spa treatments at a very up-market resort. Your main task will be to analyze the algorithmic behavior of their sea water supply system.

The resort is located at the top of a small mountain, with beautiful vistas overlooking the sea. There is a tank (Tank 0) on site that can store water for a small number n of baths. At the base of the mountain, there is another tank (Tank 1), which stores a lot more sea water—corresponding to $m \cdot n$ baths, for a positive integer m—and can be used to replenish Tank 0 when empty through a system of pumps. When Tank 1 lacks sufficient water to fully replenish Tank 0, it draws water from the sea (call this Tank 2, for convenience), which we consider of infinite capacity.

Let the current number of baths of water *missing* from Tank 0 be denoted by e_0 (and for Tank 1 the corresponding quantity is e_1). Also, let κ_0 be the cost of drawing one bath of water from Tank 0, κ_1 be the cost of pumping one bath of water from Tank 1 to Tank 0, and κ_2 be the cost of pumping one bath of water from Tank 2 to Tank 1.

(a) [6 points] You wish to find the amortized cost of drawing the water for a spa treatment bath, and you decide to develop an appropriate potential function. (Assume that initially all tanks are full to capacity.)

Which of the following potential functions is the appropriate one to use for your analysis:

(i) $\phi_a(e_0, e_1) = \kappa_1 e_0 + \kappa_2 e_1$,

(ii) $\phi_b(e_0, e_1) = \kappa_1 e_0 - \kappa_2 e_1$,

(iii) $\phi_c(e_0, e_1) = (\kappa_1 + \kappa_2) e_0 + \kappa_2 e_1$.

(No justification needed.)

(b) [24 points] Assume that we only fill a tank when empty and, in that case, we fill it to capacity. For example, when all tanks are empty and asked for a bath, we first fill Tank 1 to capacity with water from Tank 2 and then we fill Tank 0 to capacity with water from Tank 1 and then, finally, we use water from Tank 0 for the bath.

Using your answer from Part (a), prove that the amortized cost of taking a bath is equal to $\kappa_0 + \kappa_1 + \kappa_2$.

Problem 6. [25 points] **Finding Duplicates** (3 parts)
Consider the following algorithm that checks whether a given sequence $S = s_1, s_2, \ldots, s_m$ of m numbers from $\{0, \ldots, n-1\}$ has duplicates:

DUPLICATE(S)
```
Choose a random hash function h : {0, ..., n − 1} ↦ {0, ..., m − 1}
Initialize array A of size m to −1's
for j ← 1 to m
     do
        if A[h(s_j)] is equal to s_j
          then
               return "DUPLICATE FOUND"
          else
               A[h(s_j)] ← s_j
return "NO DUPLICATES"
```

(a) [10 points] Show that the probability (over the choice of h) that the above algorithm returns an incorrect answer is at most $1 - e^{-1}$.

Hint: It might be useful to consider cases when there are no duplicates and exactly one duplicate. Also, you may find the inequality $(1 - \frac{1}{k})^{k-1} \geq e^{-1}$ *useful.*

(b) [8 points] Consider the following family $\mathcal{H} = \{h_a \mid a \in \{0, \ldots, n-2\}\}$ of hash functions from $\{0, \ldots, n-1\}$ to $\{0, \ldots, n-2\}$, where

$$h_a(l) = \begin{cases} 0 & \text{if } l = n-1 \\ (a+l) \mod (n-1) & \text{otherwise.} \end{cases}$$

Is $\mathcal{H}$ universal?

(c) [7 points] Consider a version of the algorithm above in which $m = n - 1$ and we use a hash function sampled from the hash family from Part (b) instead of a fully random function.

Give an example of an input where the algorithm succeeds with probability at most $\frac{2}{n-1}$.

Problem 7. [20 points] **Car Dealer** (2 parts)
You are selling your car. However, instead of setting a fixed price, you let the clients come one-at-a-time and offer the amount of money they are willing to pay. After each offer, you have to decide whether to sell to that client or proceed to the next one (letting go of the current offer forever). Assume all the offers are between R and r, with $R \geq r > 0$, and let $\gamma = \frac{R}{r}$. (Assume you know who is the last client, and of course you can only sell the car once and so can only accept one offer.)

(a) [13 points] Design and prove correctness of a *deterministic* algorithm for this problem that, no matter what the sequence of clients (and their offers), is guaranteed to accept an offer that is within a factor of at most $\sqrt{\gamma}$ of the highest offer made in the whole sequence.

(b) [7 points] Design and analyze a *randomized* algorithm such that the *expected* value of the accepted offer is at least $\frac{1}{2\lceil \log_2 \gamma \rceil}$ of the value of the highest offer made in the whole sequence. *Hint: Randomly guess a good estimate.*

Problem 8. [35 points] **The Angels are Playing** (3 parts)
Angels Frandly and Nesty are having fun! They see you, a road engineer, trying to figure out how to build the cheapest set of roads to connect all n towns in the Kingdom of Futil. You, as a recent MIT graduate, are building roads, each with a given cost, so that any pair of towns can reach each other, and with a minimal total cost. They, being almost all powerful, can assign whatever cost they want to building a road between any pair of towns as long as the costs they choose are all different integers from the set $\{1, \ldots, \binom{n}{2}\}$. For example, if there are 6 towns, they must choose, as costs, different numbers in $\{1, ..., 15\}$.

Frandly, being a nice angel, wants to assign the costs to roads so that the overall cost of the cheapest road network is as small as possible. Nesty, being a mischievous one, wants to assign these costs so the overall cost is as large as possible.

(a) [10 points] Can you help Frandly figure out how to assign these costs to achieve her goal? Specifically, state the smallest cost of a road network that she can arrange (as a function of n), briefly argue why it is indeed the smallest cost possible, and describe how to arrange it.

(b) [15 points] Show that Nesty can assign the costs to roads so as the cheapest road network will have an overall cost of

$$\sum_{i=1}^{n-1} \left(\binom{i}{2} + 1 \right).$$

where we define $\binom{1}{2} = 0$.

(c) [10 points] *Challenging!* Prove that the overall cost bound stated in Part (b) is the maximal one possible. That is, argue that no matter how Nesty assigns costs to roads, it is always possible to build a road network whose overall cost is at most equal to that bound.