CS/ECE 374 A ✧ Fall 2026

☙ Midterm 2 Practice 1 ❧

November 5, 2026

Name:	
NetID:	

- ***Don't panic!***
- You have 120 minutes to answer five questions. **The questions are described in more detail in a separate handout.**
- If you brought anything except your writing implements, your **hand-written** double-sided 8½" × 11" cheat sheet, and your university ID, please put it away for the duration of the exam. In particular, please turn off and put away *all* medically unnecessary electronic devices.
- Please clearly print your name and your NetID in the boxes above.
- Please also print your name at the top of every page of the answer booklet, except this cover page. We want to make sure that if a staple falls out, we can reassemble your answer booklet. (It doesn't happen often, but it does happen.)
- Greedy algorithms require formal proofs of correctness to receive credit. Otherwise, proofs or other justifications beyond those listed in the standard grading rubrics are required if and only if we explicitly ask for them, using the word ***prove*** or ***justify*** in bold italics.

- **Do not write outside the black boxes on each page**. These indicate the area of the page that our scanners will actually scan. If the scanner can't see your work, we can't grade it.
- If you run out of space for an answer, feel free to use the overflow/scratch pages at the back of the answer booklet, but **clearly indicate where we should look**. If we can't find your work, we can't grade it.
- **Only work that is written into the stapled answer booklet will be graded**. We **do not** recommend detaching pages from the answer booklet, because any work on those detached pages will not be graded (and it disrupts our process for uploading solutions to Gradescope.) Please let us know if you detach a page either on purpose or accidentally so we can reattach it.
- We will provide additional blank scratch paper on request, but we will not grade anything written on it.
- **Please return all paper to Emily or a TA** including your **answer booklet**, your question sheet, your cheat sheet, and all scratch paper. There will be two piles of collected paper, the first for answer booklets and the second for everything else. We **will not** consider the paper in the second pile when grading your exam.

This sentence contains four and three fourths percent a's, five and one half percent c's, three percent d's, eighteen and one fourth percent e's, four and one half percent f's, three fourths percent g's, five percent h's, two and one half percent i's, two percent l's, twelve and one half percent n's, six percent o's, five percent p's, eight percent r's, seven percent s's, nine and three fourths percent t's, two and one fourth percent u's, one and one half percent v's, one and one fourth percent w's and one half percent x's.

CS/ECE 374 A ✧ Fall 2026 Midterm 2 Practice 1 Problem 1	Name:

(a) Write the solution to each of the following recurrences in the box immediately below it. (Use the space below the boxes for scratch work.)

$A(n) = A(5n/11) + O(\sqrt{n})$ $B(n) = 8B(n/2) + O(n^2)$ $C(n) = C(n/2) + C(n/3) + C(n/6) + O(n)$

(b) Describe an appropriate memoization structure and evaluation order for the following (meaningless) recurrences, and state the running time of the resulting iterative algorithm to compute the requested function value.

- Compute $Foo(1, n)$ where

$$Foo(i,k) = \begin{cases} 0 & \text{if } i \geq k-1 \\ \max\left\{ \begin{matrix} Foo(i,j) \\ \quad + Foo(j,k) \end{matrix} \;\middle|\; i < j < k \right\} + \sum_{j=i}^{k} A[j] & \text{otherwise} \end{cases}$$

- Compute $Bar(n, 1)$ where

$$Bar(i,s) = \begin{cases} \infty & \text{if } i < 0 \text{ or } s > n \\ 0 & \text{if } i = 0 \\ \min\left\{ \begin{matrix} Bar(i, 2s), \\ X[i] \cdot s + Bar(i-s, s) \end{matrix} \right\} & \text{otherwise} \end{cases}$$

CS/ECE 374 A ✧ Fall 2026 Midterm 2 Practice 1 Problem 2	Name:

Quadhopper is a solitaire game played on a row of n squares. Each square contains four positive integers. The player begins by placing a token on the leftmost square. On each move, the player chooses one of the numbers on the token's current square, and then moves the token that number of squares to the right. The game ends when the token moves past the rightmost square. The object of the game is to make as many moves as possible before the game ends.

(a) ***Prove*** that the obvious greedy strategy (always choose the smallest number) does not give the largest possible number of moves for every quadhopper puzzle.

(b) Describe and analyze an efficient algorithm to find the largest possible number of legal moves for a given quadhopper puzzle.

CS/ECE 374 A ✧ Fall 2026 Midterm 2 Practice 1 Problem 3	Name:

After moving to a new city, you decide to walk from your home to your new office. To get a good daily workout, you want to reach the *highest* possible altitude during your walk (to maximize exercise), while keeping the total length of your walk below some threshold (to get to your office on time). Describe and analyze an algorithm to compute the best possible walking route.

Your input consists of an undirected graph G, where each vertex v has a height $h(v)$ and each edge e has a positive length $\ell(e)$, along with a start vertex s, a target vertex t, and a maximum length L. Your algorithm should return the *maximum height reachable* by a walk from s to t in G, whose total length is at most L.

CS/ECE 374 A ✧ Fall 2026 Midterm 2 Practice 1 Problem 4	Name:

A ***verbal subsequence*** of a string T is a sequence of non-overlapping substrings of T, each of which is a word. The *length* of a verbal subsequence is the number of words it contains. Describe and analyze an algorithm to find the length of the longest verbal subsequence of a given string $T[1..n]$. You have access to a black-box subroutine IsWord that can decide whether an arbitrary string w is a word in $O(|w|)$ time.

CS/ECE 374 A ✧ Fall 2026 Midterm 2 Practice 1 Problem 5	Name:

Describe and analyze an algorithm to compute the Euclidean distance between the closest pair of points in a given set P of n points in the plane, using subroutines IWalkTheLine and Distance as black boxes.

(See the question handout for definitions of the black-box subroutines.)

(scratch paper)

(scratch paper)

(scratch paper)

(scratch paper)

(scratch paper)