

Midterm Examination

CS 362 Data Structures and Algorithms
Spring, 2026

Name:
Email:

Directions:

- This exam lasts 75 minutes. It is closed book and notes, and no electronic devices are permitted. However, you are allowed to use 2 pages of “cheat sheets”
 - *Show your work!* You will not get full credit, if we cannot figure out how you arrived at your answer.
 - Write your solution in the space provided for the corresponding problem.
-

Question	Points	Score	Grader
1	20		
2	20		
3	20		
4	20		
5	20		
Total	100		

1. **Short Answer (4 points each)**

Answer the following using *simplest possible* Θ notation.

(a) Expected number of items at the $\log n$ -th level (from the bottom) of a skip list containing n items?

(b) Solution to the recurrence: $T(n) = 8T(n/3) + n^3$

(c) Solution to the recurrence: $T(n) = 8T(n/2) + \sqrt{n}$

(d) Solution to the recurrence: $f(n) = 7f(n - 1) - 10f(n - 2) + 2^n$
(answer in big-O)

(e) Solution (in big-O) to the recurrence $F(n) = 2F(\sqrt{n}) + \log n$.

2. Induction (20 points)

A **Sandia String** consists of (P)ink and (V)iolet characters as follows:

- The string consisting of the single character P is a Sandia String.
- If A , B , and C are Sandia Strings, then the string $VAVBVCV$ is also a Sandia String.

For example, if we use the base case for A , B , and C , the string $VPVPVPV$ is a Sandia String.

Let $p(S)$ be the number of P's and $v(S)$ be the number of V's in Sandia string S . Prove that for any Sandia String S , $v(S) = 2P(S) - 2$. Prove this by induction on the length of the string, n . Don't forget the BC, IH, and IS.

3. Sandia Tram

You are riding the Sandia Tram, which passes by n scenic checkpoints on its way to the peak. At each checkpoint i (where $1 \leq i \leq n$), you have two choices to capture the view:

- **Take a Photo:** This is fast setup and gives you a “joy” score of p_i .
- **Record a Video:** This gives you a higher “joy” score of v_i . However, it takes time to setup your camera for video, meaning that you cannot do anything at checkpoint $i - 1$.

Your goal is to maximize your total joy score over the n checkpoints. Assume you are given all of the p_i and v_i values for $1 \leq i \leq n$ and your goal is to return the maximum joy score achievable.

- (a) (2 points) Describe in words a function whose solutions for smaller problems will help you solve the big problem.
- (b) (14 points) Write a recurrence relation for this function. Don't forget the base case(s).

(c) (4 points) Describe a dynamic program to solve this problem. What are the dimensions of your table? How do you fill it? What is the final value returned? What is the runtime of your algorithm?

4. Probability and Expectation (20 points)

During the Balloon Fiesta, n hot air balloons take off over k days.

- (a) (4 points) On day 1, each of the n balloons independently has a probability $p = 3/n$ of flying directly over Farris. What is the expected number of balloons that fly over Farris?
- (b) (4 points) Use Markov's inequality to get an upper bound on the probability that 9 or more balloons fly over Farris on day 1.
- (c) (4 points) The Fiesta lasts for k days. The probability of severe weather canceling the launch on any given day is $1/k^3$. Use a union bound to upper bound the probability that severe weather cancels the launch on *at least one* of the k days.

(d) (8 points) For the mass ascension, the n balloons are assigned uniformly at random to n different launch sites. This means each balloon independently chooses one of the n sites with probability $1/n$. What is the expected number of *empty* launch sites (sites with zero balloons assigned to them)?

5. **Balloons.** You are given n balloons, each with a number painted on it. You are asked to pop all the balloons. The numbers on the balloons are started in a *nums* array of length n . If you pop the balloon at index i ; and ℓ is the index of the closest un-popped balloon to the left; and r is the index of the closes un-popped balloon to right, you get a number of coins equal to $nums[\ell] * nums[i] * nums[r]$. If ℓ or r goes out of bounds of the array, then treat it as if there is a balloon with a 1 painted on it. Your goal is to return the maximum coins you can collect by bursting the balloons in a smart order.

(a) (2 points) Consider the following naive algorithm. In each step, you pop the balloon that gives you the largest number of coins for that step. Give an example input showing that this algorithm may not maximize the total number of coins.

(b) (2 points) To set up a dynamic program, define in words a smaller problem(s) whose solutions can help you solve the big problem. Please define this problem as a function and specify the arguments to it.

(c) (14 points) Now write a recurrence relation for a dynamic program to solve this problem. Hint, it may help to set $nums[0]$ and $nums[n+1]$ to 1.

(d) (2 points) Describe your dynamic program and give the runtime.