
Midterm 1

This midterm is due **September 16 at 11:59PM**. The suggested lengths of explanations are exactly that: suggestions. You do not need the exact specified number of sentences. These suggestions exist solely to provide an idea for the expected level of detail.

Problem 1-1. [20 points] Correctness

Determine whether each of the following algorithms is fully correct, and prove that your answer is correct.

(a) [10 points]

$A(x)$

```
IN :  $x \in \mathbb{R}, x \geq 1$ 
1  $x' = x$ 
2  $p = 0$ 
3 while  $x' > 1$  do
4    $p = p + 1$ 
5    $x' = x'/2$ 
6 return  $p$ 
OUT:  $p = \log_2 x$ 
```

(b) [10 points]

$B(x)$

```
IN :  $x \in \mathbb{R}, x \geq 1$ 
1 while  $x > 1$  do
2    $x = x/2$ 
3 if  $x == 1$ 
4   return true
5 else
6   return false
OUT: true if  $x = 2^n$  for some integer  $n$  (initially); false otherwise.
```

Problem 1-2. [20 points] Data Structure Selection

(a) [10 points]

Imagine you are implementing a graph search mechanism; you want to find the shortest path from some *source* node to a *target* node. You decide to use Dijkstra's Algorithm. This algorithm requires a `PriorityQueue`. The `PriorityQueue` must be able to do the following in $O(\log n)$ time (where n is the number of elements):

- `insert(key, value)`
- `extract_min`
- `decrease_key(old_key, value, new_key)`

What data structure should be used to implement the `PriorityQueue` ADT in this case? Justify your answer.

(b) [10 points]

You've joined the development team for an indie game called **Amidst Us**. Each player can see how many games they've won on the home screen; this number is calculated as follows:

- Each time a player wins a game, their username is added to the end of a list.
- Each time a player enters the home screen (so their number of wins needs to be calculated), this list of usernames is traversed and the number of occurrences of the player's username is counted.

Recently, the number of players playing **Amidst Us** has skyrocketed after several influential streamers picked it up. Users are now complaining that the home screen takes several minutes to load (and the problem is getting worse, fast).

You're tasked with fixing the problem. You've determined that the problem is caused by the calculation discussed above. What structure should you use to store and easily increment the number of wins associated with each username in order to speed up this process? Justify your answer.

Problem 1-3. [20 points] Sorting Lower Bounds

- (a) [10 points] Explain briefly (2–3 sentences) how we know that sorting must take $\Omega(n \log n)$ time in the comparison model.
- (b) [10 points] `RadixSort` can sort integers of bounded size in linear time. Does this contradict the result discussed in the previous part? Why or why not?

Problem 1-4. [20 points] **Dictionary Management**

Determine whether each of the following dictionary resizing schemes will maintain $O(1)$ insert and delete amortized complexity. The costs of maintaining dictionary size should be included in the costs of the operations. Justify each answer briefly. In each part, n is the number of elements and m is the size of the dictionary.

- (a) [10 points] Every time an element is inserted or deleted, resize the dictionary so that $m = 2n$.
- (b) [10 points] After inserting, if $n > \frac{m}{2}$, double the size of the dictionary. After deleting, if $n \leq \frac{m}{4}$, halve the size of the dictionary.

Problem 1-5. [20 points] **Optimization Overview**

Briefly describe (in 3–4 sentences) the series of steps taken to optimize (or attempt to optimize) a slow implementation. Where do you start? How do you proceed?