Practice Midterm Exam (Modules 1-7)
Comp 582 Rice University Fall 2022

Instructions

For the exam, you may refer to videos, video slide sets, problem sets, live session slides, and any of your personal notes or materials. You may NOT, however, collaborate with any other people. You may also use the Internet in a limited way. You may search for something on the Internet, but you may NOT ask a question on any question answering site (like Stack Overflow). If you use something you found on the Internet, please cite whatever you found.

For short answer questions, you may just write the answer.

For long answer questions, you must justify *all* aspects of the problem — even if the problem does not explicitly ask for justification or proof.

When justifying your answer, please show all relevant work, and *only* relevant work. If you turn in spurious work, or elements of approaches you discarded, I will assume that you think whatever you wrote should contribute to the solution. Anything you turn in will be assumed to be part of your answer.

When you have finished your exam, please upload to Canvas (just like the problem sets).

You may, if you wish, email your instructor a "just-in-case..." copy of your exam.

Format

There are <SOME> quick answer problems, and <SOME> "not-so-quick" problems on the exam.
(The real exam will list exact numbers)

As with your problem sets, you may turn in PDF or plain text. The PDF can be scans of handwritten material.

When Giving an Algorithm

The code standard for this exam is the same as the standard for problem sets. Namely, you **must** use Restricted Python. The restrictions are:

- No Python dictionaries (or classes derived from dictionaries)
- No Python sets (or classes derived from sets)

You may use the python `heapq` module if you want.

Time Limit

You have a time limit of 4 hours.

Pledge

Do not forget to put the Honor Code Pledge on your last age of your exam (you may use a separate page if necessary). Please be sure to include your start time and end time as part of the Pledge.

Conventions and Nomenclature

Throughout this exam, if the question asks for "complexity", the correct answer will be the worst-case time complexity of the best algorithm you know to solve the problem.

Unless otherwise stated, any answer involving complexity, the Big Oh notation is sufficient.

In any problem text that refers to "sorting", the sort order is presumed to be ascending order unless otherwise stated.

Quick Answer

1. What is the worst-case complexity for any single union operation when using weighted quick union and path compression?

2. What is the worst-case complexity for a series of M union/find operations on n (initially) disjoint sets when using weighted union and path compression?

3. What is the worst case for series of operations on dynamically-sized arrays? NOTE: The series of operations can contain pushes.

For the next 5 questions, you are given 2 functions of n, $f_1(n)$, $f_2(n)$. Decide if $f_1(n) = O(f_2(n))$, or $f_2(n) = O(f_1(n))$, or both.

4. $f_1(n) = x^{1000}$, $f_2(n) = 1.00001^n$

5. $f_1(n) = \log(n)$, $f_2(n) = n$

6. $f_1(n) = n$, $f_2(n) = \log^3(n)$

7. $f_1(n) = n!$, $f_2(n) = n^n$

8. $f_1(n) = \log(n!)$, $f_2(n) = n \log(n)$

9. What is the worst case complexity for selection sort n items?

10. What is the best case complexity for selection sort of n items?

11. What is the worst case complexity for insertion sort n items?

12. What is the best case complexity for insertion sort of n items?

13. Worst-case for Quicksort of n items?

14. Average case for quicksort of n items?

15. Worst-case for merge sort n items?

16. Average case for merge sort items?

17. Worst case for quickselect to find the n/2-smallest item in a set of n items?

18. Average case for quickselect to find the n/2-smallest item in a set of n items?

19. Worst case for quickselect to find the minimum item in a set of n items?

20. Average case for quickselect to find the minimum item in a set of n items?

21. What is the best algorithm for finding the minimum item in a set of n items?

Suppose you are looking for an item with key k in a set of n unsorted items.

22. What is the worst-case complexity to find the item (assuming it is in the set)?

23. What is the average-case complexity to find the item (assuming it is in the set)?

24. What is the worst-case complexity to determine the item is not in the set?

25. What is the average-case complexity to determine the item is not in the set?

Suppose you are looking for an item with key k in a set of n sorted items.n

26. What is the worst-case complexity to find the item (assuming it is in the set)?

27. What is the average-case complexity to find the item (assuming it is in the set)?

28. What is the worst-case complexity to determine the item is not in the set?

29. What is the average-case complexity to determine the item is not in the set?

30. For Shellsort, let the h-gap sequence be {1, 4, 9}
    Let the input sequence be

    x, y, z, a, b, c, d, e, f, g

    Show the sequence after the $h = 9$ gap is completed

31. What is the worst-case complexity of shellsort using the h-gap sequence
    $\{1,2...,2^{i-1}, ...\}$?

32. What is the worst-case complexity of shellsort using the h-gap sequence
    $\{1,4,...,3^i + 1,...\}$

33. What is the complexity of a KFY shuffle? (Assume getting a random number is
    $O(1)$)

34. What is the worst-case complexity of a sort shuffle (Getting a random number is O(1))?

35. In a heap of size N, how many elements are leaves?

36. For a 0-based array, what are the formulas for the left child, right child and parent?

37. 'In the Union/Find algorithm that uses weighted quick union, what is the worst case height of any tree in the forest of sets?

Not-so-Quick Answers

1. Is the following (pseudo) code for program A an algorithm? Justify your answer.

```
A(x[]) {
    i=0;
    while (external_test(i)) {
        x[i] = 0;
        i = i+1;
    }
}
```

2. What is the computational complexity of the following program? (Big Oh notation is sufficient). Assume that `X[i,j]` references the `(i,j)` component of matrix `X`. Also, assume that `N,M` are parameter describing the size of `X` as `X[N,M]`.

```
for (i=1;i<=N;i++){
    v[i] = 0;
    for (j=i;j<=M;j+=2) {      # (M - i + 1)//2
        v[i] += A[i,j]*b[j];
    }
})
```

3. Suppose that we choose 4 as our factor in the "Big Box Store" algorithm. That is, when the stack is full we allocate 4x the amount of current storage. What is the complexity of a sequence of M pushes and array refs?

4. Draw the tree associated with the f array below:

```
i   | 1  2  3  4  5  6  7  8  9  10
    +————————————————————————————————
f[i] | 2  2  4  2  6  7  2  4  5  6
```

Could weighted quick union have been used to construct the f array shown?

5. Show with a sample trace how insertion sort sorts the letters

   A Q U E S T I O N

6. Show with a sample trace how shellsort with the h-sequence 10,5,2,1 sorts the letters

   V E R Y L O N G E N T R Y

7. Using any method you like, find the Big Oh complexity of the recurrence

$$T(n) = 3T\left(\frac{n}{2}\right) + 2n^2 + 17n + 19$$

8. Using any method you like, find the Big Oh complexity of the recurrence

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + 1500n + 17\log(n) + 4000001$$

Define the <u>pmedian</u> of a set of n items as

$$\text{pmedian}(S) = \begin{cases} \text{median}(S) & \text{if } |S| \text{ is odd} \\ \max_{x \in S} \ x \leq \text{median}(S) & \text{if } |S| \text{ is even} \end{cases}$$

9. Suppose you could always find the pmedian for a set of n items in O(1) time. What would be the worst case performance for quicksort if you pivot on the pmedian?

10. Suppose you could always find the pmedian of a set of n items in O(n) time. What would be the worst case performance for quicksort if you pivot on the pmedian?