

## CMSC 341 (Practice) Midterm Exam 2 (Fall 2018)

Instructions: Clearly write your name on this sheet. Answer each problem in the space provided. If you need extra space, clearly write your name and the problem number on an extra sheet of paper, write “on extra sheet” in the answer space on the exam paper, and turn in the extra sheet with your exam.

**Write legibly.** If the person grading the test cannot read something, s/he will simply assume that you meant the illegible portion as a note to yourself and they will ignore it. If you lose points because part of your answer could not be read, you will not be given the opportunity to explain what it says.!

**Be clear and concise.** The answers to most questions should be short. If you find yourself writing an excessively long response, you may want to think more carefully about the question. Long rambling answers generally get fewer points than short ones do because there are more opportunities to mark something wrong. **The purpose of short-answer questions is to determine if you know what you are talking about, a few key words can usually communicate this.**

You may not ask questions of other students, look at another student’s exam, use a textbook, use a phone or calculator, or seek any other form of assistance. In summary: do not cheat. Persons caught cheating will be subject to disciplinary action.

Each question is marked with a number of points. There are 100 points total.

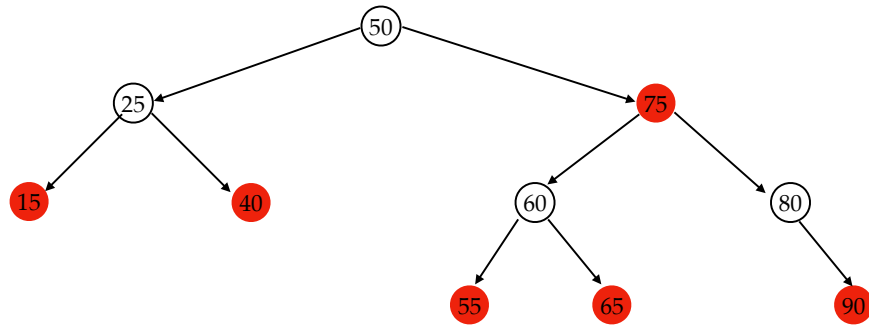
If something isn’t clear, ask!

Good luck

1. (2 pts) **Name:** \_\_\_\_\_

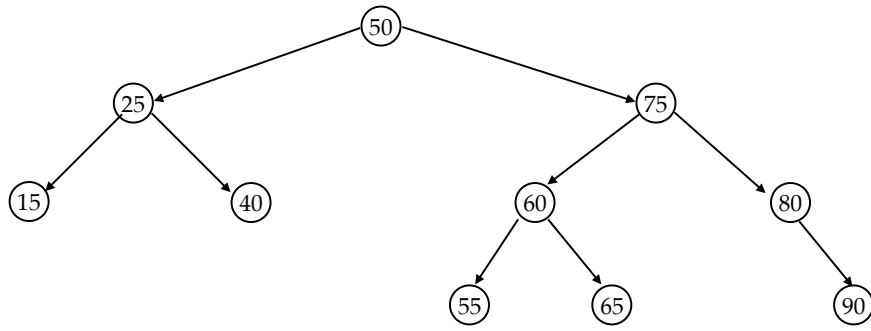
2. (3 pts) **GL Login:** \_\_\_\_\_

**DO NOT OPEN THE TEST UNTIL TOLD TO BEGIN**



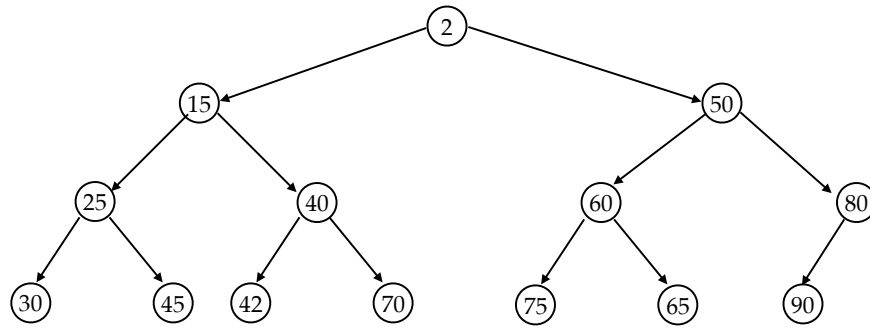
**Figure 1:** A red-black tree.

3. (10 pts) Insert 70 into the red-black tree in Figure 1.



**Figure 2:** A *splay tree*.

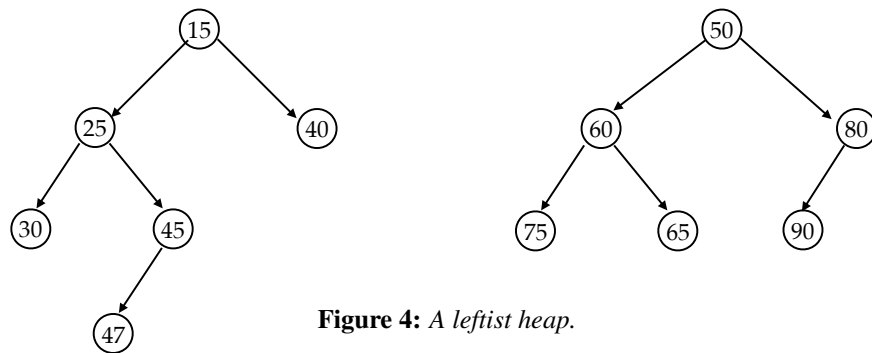
4. (10 pts) Insert 70 into splay tree in Figure 2.



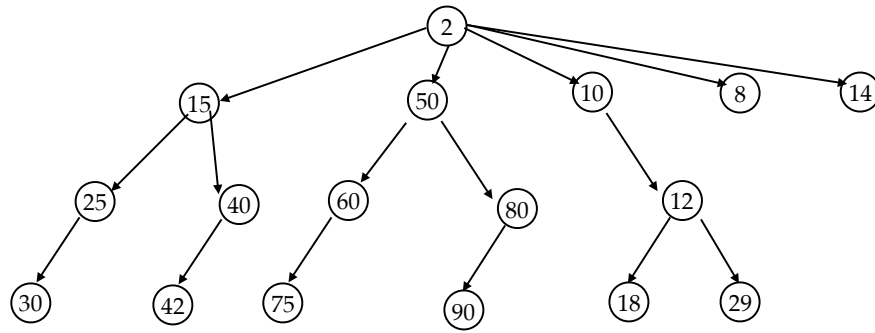
**Figure 3:** *A binary heap.*

5. (5 pts) Show how the heap in Figure 3 would be stored in an array.
6. (10 pts) Insert 5 into heap in Figure 3.
7. (10 pts) Remove the minimum from the heap in Figure 3.

8. (10 pts) Run the linear-time *Heapify* algorithm on the following array of numbers: 15, 32, 18, 14, 2, 9, 3, 20, 10, 5, 25, 30, 16, 4, 8.



9. (10 pts) Merge the two leftist heaps in Figure 4.



**Figure 5:** *A pairing heap.*

10. (5 pts) Insert 16 into the pairing heap in Figure 5.
11. (10 pts) Delete the minimum from the pairing heap in Figure 5.

12. (5 pts) Insert the following numbers into a hash table of size 13 using linear probing: 17, 19, 5, 30, 31.
13. (5 pts) Insert the following numbers into a hash table of size 13 using quadratic probing: 17, 19, 5, 30, 31.
14. (5 pts) Insert the following numbers into a hash table of size 13 using separate chaining: 17, 19, 5, 30, 31.