CSC190: Test 2019-04-17

First name (please write as legibly as possible within the boxes)

Last name

Student ID number

Instructions

- You are advised to use pen. Pencil is NOT recommended for this exam. Don't detach the pages.
- Write your official first name, last name, student number using UPPER-CASE letters and numbers.
- Closed book; no aids; no electronic equipment allowed (cellphones, tablets, computers, calculators, etc.).
- Attempt and answer all questions on all pages.
- No questions will be answered by the examiner: if a particular question seems unclear, explicitly state any reasonable assumptions and proceed with the problem.
- Show all steps and present solutions clearly using only concepts discussed in lectures.

1. a) (5 points) Define a pointer, g, that can point to a function that returns an integer when provided with two float arguments.

b) (5 points) Define a pointer, f, that can point to a function that returns an integer when provided with two arguments as follows:

- the first argument is a function of the same type as what g (from the prior part of this question) points to
- the second argument is a float

c) (5 points) Show how you would call the aforementioned f when provided the aforementioned g and 3.14 as arguments

d) (5 points) Define a pointer, h, that can point to a function that returns a struct blah **** and takes as arguments a pointer to a struct blah ******* and an int.

2. (10 points) Consider unsigned int x, and unsigned chars a,b,c,d. Show C code to assign a,b,c and d, as follows:

   - a should contain the bits of x whose positions are divisible by 4
   - b should contain the bits of x whose positions when reduced by 1 are divisible by 4
   - c should contain the bits of x whose positions when reduced by 2 are divisible by 4
   - d should contain the bits of x whose positions when reduced by 3 are divisible by 4

   Do not change the left to right order of bits, just positions; recall the left most position is 31 and the rightmost is 0 for an unsigned int.

3. (10 points) Consider the language, variables and definitions of the prior question; given unsigned int y (which may have valid data), assign to y as follows:

   - bits 31-24 of y are the bits of a
   - bits 23-16 of y are the bits of b
   - bits 15-8 of y are the bits of c

   Do not alter the left to right order of bits; bits 7-0 of y must not be changed.

4. **(10 points)** Complete **all** empty cells of the following table. The values in each column are displayed in the base indicated by the column heading.

decimal (base-10)	hexadecimal (base-16)	binary (base-2)
992		
	0x1be	
		1001011110
	0x48c	
301		

5. For the following questions on time complexity, you must:

   - circle the big-O time complexity that you are reporting as your answer
   - indicate your reasoning

   Where needed, assume that all trees, lists, etc., have n elements.

   - **(5 points)** F is a function that takes a list as input (size N), where the value at each node is a binary tree (size M) of integer elements. Let's say the tree product of N trees is defined as the sum of the products formed by multipling corresponding elements of the lists obtained by pre-order traversals of said trees. What is the big-O time complexity of F, if F takes the tree product of the trees found in its argument list?

   - **(5 points)** F is a function that takes a binary tree (size N) as input, where the value at each node is a list of integers (size M) F traverses the tree, mergesorts each list element, and returns the concatenation of all lists found in the tree. What is the big-O time complexity of this function?

- **(5 points)** F is a function that takes a sorted list (whose sort direction is known) as input and returns the value of the average of the list. What is the big-O time complexity of this function?

- **(5 points)** F is a function that takes a binary tree as input, where the value at each node of the tree is an list (ascendingly sorted) of integers. F traverses the tree, and returns the sum of the largest element in each list visited in the traversal. What is the big-O time complexity of this function?

- **(5 points)** F is a function that takes a binary tree (size K) as input, where the value at each node is a list of integers (size M) F returns the tree of lists, but with all of the lists bubble sorted. What is the big-O time complexity of F?

- **(5 points)** For the prior question: what is the fastest sorting algo (excluding radix sort) we discussed in this course, and using that what is the big-O time complexity of F?

- **(6 points)** What is the time complexity of a function that takes three $n \times n$ matrices as input (A,B,C) and returns A*B*C? To answer this, consider only iterative approaches to matrix operations that a child in grade school would perform by hand.

- (5 points) The number, *cake*, is one that has fascinated pseudo-scientists for aeons. To calculate *cake*, via algorithm A, it is known that the number of steps required to calculate digit i is twice the number of steps to calculate digit $i-1$. What is the big-o complexity of algorithm A for N digits?

- (5 points) (Still thinking of *cake*). To calculate *cake*, via algorithm B, it is known that the number of steps required to calculate digit i is half the number of steps to calculate digit $i-1$. What is the big-o complexity of algorithm B for N digits?

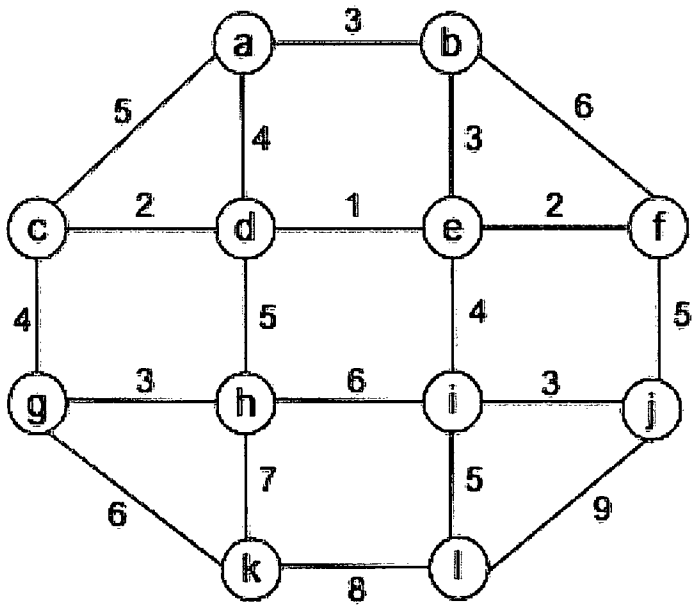
- (5 points) (More *cake*). To calculate *cake*, via algorithm C, it is known that the number of steps required to calculate digit i is one less than the the number of steps to calculate digit $i-1$. What is the big-o complexity of algorithm C for N digits?

6. Consider the following relationships: A+B, B+C, C+P, P+Q, Q+R, R+P, R+A, A+K, K+L, L+M, L+F, M+K, M+C, B-F, F-C, where the + operator means the left operand has a relationship with the right operand (but not vice versa), while the - operator means the two operands are reciprocally related. **(15 points)** List the vertices that would be visited by a breadth-first traversal of this graph starting at A. When considering multiple vertices at the **same** level that you need to visit, choose the vertex with the lowest alphabetical value first.

7. **15 points** Show clearly how heap sort will sort the list Y, C, A, X, G, F, K, E, D in descending order (note: $A < Z$). Show the state of each step of the algorithm, annotating what is occuring with each step; you may assume that heapify and reheapify are atomic operations (i.e., you don't have to delve into their steps).

8. **15 points** Redo the prior question using merge sort, and showing all steps using the presentation mechanism discussed in class.

9. **5 points** For graph shown in the figure, show the adjacency list representation.



10. **10 points** Show the adjacency matrix representation for the prior graph.

11. (20 points) An stack is fed the following data a,b,c,g,d,h,e,f in an order such that item f was the last element to be fed (do not reorder elements beyond this).

    a) (2 points) Show the stack once fully fed, indicating where the head is.

    b) (14 points) Pop the elements from the stack, and as each element is popped, insert them into an AVL tree (alphabetical order, such that $a < b$, etc.). Show all steps: starting from an empty AVL tree, show how the tree diagram looks like: before each item is added, after the item has been added but before the AVL tree is restored, and after the item has been added and the AVL property restored. Annotate all non-leaf nodes with the AVL balance for the diagrams you draw (left heavy is negative, right heavy is positive).

c) For the prior AVL tree, indicate the (2 points) pre-order and (2 points) post-order traversals of the tree.

12. (5 points) You have a sensor that outputs integer data in the range $[-65, +120]$, but you want this data to occupy an integer range that starts at 1000, and where consecutive data values (from the original set) are separated (in the new desired set) by unused values. So for example, if your sensor reported -65, you would want to modify that report to say 1000. And -64 would say 1002 (see how 1001 was skipped?). You can't modify the sensor, but you can modify the data: how would you modify this data using simple integer operations (no floats). Write your answer as a C function called convert, which takes in an int and returns an int.

13. **15 points** An queue is fed the following data a,b,c,g,d,h,e,f in an order where item f was the first element to be fed in (re-read this sentence).

    a) (1 points) Show the queue once fully fed, indicating where the head and tail are.

    b) (14 points) Dequeue the elements from the queue, and as each element is dequeued, insert them into a straight, unadulterated BST tree (alphabetical order, such that $a < b$, etc.). Show all steps: starting from an empty BST tree, show how the tree diagram looks like: before each item is added, and after the item has been added.

14. **20 points** Consider the following graph relations: $(A/B, 1)$, $(A/C, 10)$, $(C/B, 5)$, $(B/D, 2)$, $(C/D, 5)$, $(C/E, 2)$; $(\sigma/\epsilon, \xi)$ means that the edge connecting σ and ϵ has weight ξ. Showing all steps clearly, skipping nothing, perform Dijkstra's shortest path algorithm on this graph starting from E. Where relevant, use u and ∞. Clearly indicate your final answer (the output of Dijkstra's algorithm).

(Rough work; no solution should appear here)

(Rough work; no solution should appear here)