Fall 2021 COP 3502 Section 2 Final Exam - Part A

Last Name: _____, First Name: _____

1) (10 pts) For a particular video game, each user has a name (letters only) and a number of points (non-negative integer less than 10^9). Imagine reading in from standard input, information about n users and storing that information in an **array of pointers to user**, where user is the struct shown below. The format of the information being inputted is as follows:

1. First line has a single positive integer, *n*, representing the number of users.
2. Each user follows, with the information for each user being on three lines.
    a. The first line for the i^{th} user has a single positive integer L_i, representing the length of the i^{th} user's username

    b. The second line for the i^{th} user has a string of L_i letters, representing the name of the i^{th} user.

    c. The third line for the i^{th} user has an non-negative integer, p_i, representing the number of points the i^{th} user has.

Here is a short sample input:

```
4
13
SUPERCRAZYMAN
2000
11
WonderWoman
6789
17
JakeFromStateFarm
12
18
FloFromProgressive
49
```

Complete the function below to read in this information and return a pointer to an array of pointers to struct storing this information. The struct storing a user is as follows:

```
typedef struct user {
    int nameLen;
    char* name;
    int points;
} user;
```

In particular, your function is responsible for dynamically allocating all the necessary memory AND reading in the information into the array of pointers to struct and returning a pointer to the resulting array. **Make sure to allocate exactly the necessary space for the pointer name for each user struct.**

In order for this function to be useful, a pointer to the variable storing the size of the array (first piece of information being read in) is the only input parameter to the function. Please read the first item into the variable pointed to by this pointer.

```
user** readUsers(int* ptrNumUsers) {









}
```

2) (8 pts) An arithmetic sequence is a sequence of numbers such that the difference between each pair of successive terms it the same. For example, 4, 7, 10, 13, and 16 is an arithmetic sequence of length 5 with a starting term of 4 and a common difference of 3. Write a **recursive function** which takes in the first term in an arithmetic sequence, the common difference of the sequence and the number of terms in the sequence and returns the sum of the sequence.

```
int sumArithSeq(int firstTerm, int diff, int nTerms) {









}
```

3) (10 pts) The following function solves a problem, assuming that the input array is sorted:

```
// Pre-condition: sortedArray is a sorted from smallest to
//      largest, and has n values.
int whatDoesItDo(int* sortedArray, int n, int value) {

    int a = 0, b = n-1;
    while (a < b) {
        if (sortedArray[a]+sortedArray[b] == value) return 1;
        if (sortedArray[a]+sortedArray[b] < value)
            a++;
        else
            b--;
    }

    return 0;
}
```

(a) (4 pts) As specifically as possible, in English, explain what problem this function solves. Any explanation which tries to convert the code literally (then add one to a, etc.) will get **NO CREDIT.**

(b) (4 pts) If the input array isn't sorted, then this function will NOT properly solve the problem designated in part (a) in all cases. Give a specific case (array, length of the array and value) where the solution to the problem is 1 but the function as its written returns 0.

(c) (2 pts) What is the run time of this function, in terms of the input parameter, n? Briefly justify your answer.

Fall 2021 COP 3502 Section 1 Final Exam - Part B

Last Name: _____, First Name: _____

4) (10 pts) Imagine storing a string of letters in a linked list and building a set of string functions for this type of storage. Complete the function strCompare below which should operate the exact same way that strcmp behaves in the string class. Namely, if the string represented by the linked list pointed to by str1 comes lexicographically before the string represented by the linked list pointed to by str2, return a negative integer. If the string represented by the linked list pointed to by str1 comes lexicographically after the string represented by the linked list pointed to by str2, return a positive integer. If the strings are equal, return 0. The struct definition is provided below, along with the function prototype. You may assume that all characters are lowercase letters for ease of implementation. **(Hint: For ease of implementation, write recursively!)**

```
#include <stdio.h>
#include <stdlib.h>

typedef struct str {
    char letter;
    struct str* next;
} str;

int strCompare(str* str1, str* str2) {




}
```

5) (12 pts) Solve the following recurrence relation (solve for a direct solution to T(n) in terms of a Big-Oh bound) in two ways: (a) Use the Master Theorem, (b) Use the Iteration Technique. (Note: Knowing what the end answer should be via step (a) should help you discover mathematical errors in part (b).)

$$T(n) = 3T\left(\frac{n}{2}\right) + n^2 \text{ (if n > 1)}$$
$$T(1) = 1$$

(a) Use Master Theorem

(a) Use Iteration Technique

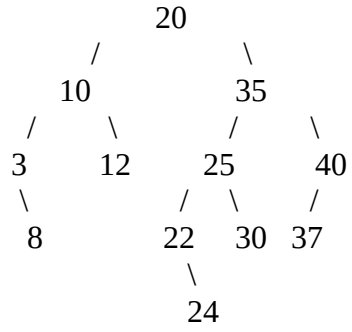
Fall 2021 COP 3502 Section 1 Final Exam - Part C

Last Name: _____, First Name: _____

6) (5 pts) Consider tracing through a Merge Sort of the following array. Show the state of the array after each merge, in the order in which the merges occur. For clarity, the result after the first merge and the result after the last merge are included.

Initial Array	12	3	9	8	16	13	2	14
After 1st Merge	3	12	9	8	16	13	2	14
After 2nd Merge								
After 3rd Merge								
After 4th Merge								
After 5th Merge								
After 6th Merge								
After 7th Merge	2	3	8	9	12	13	14	16

7) (7 pts) Show the result of deleting 12 from the AVL Tree shown below. Put a box around your final answer.



8) (10 pts) Write a function that takes in a pointer to a binary tree node which is the pointer to the root of a **binary search tree** and prints out the values stored **in the leaf nodes** in **reverse sorted order**, from largest to least. (Hint: The code represents an edit to one of the usual tree traversals, with an extra base case for printing.) Please use the struct provided below.

```
typedef struct treenode {
    int data;
    struct treenode* left;
    struct treenode* right;
} treenode;

void printLeafNodesReverse(treenode* root) {









}
```

Fall 2021 COP 3502 Section 2 Final Exam - Part D

Last Name: _____, First Name: _____

9) (5 pts) Convert 34564_7 to base 9.

_____

10) (6 pts) For accreditation, all computer science students in a class are tested on 20 skills, numbered 0 through 19. The accreditation committee wants to the number of skills that ALL of the students have successfully completed. The first input to the function is an array containing integers, all in between 0 and 2^{20}-1, where each integer represents the skills of a single student. The second input to the function is the length of the input array. In particular, if bit k is set to 1 in the value student[i], then that means that student i has completed skill i. Complete the function below to complete the task. You may call the countBits function provided in your function.

```
int numSkillsAllHave(int student[], int n) {









}

// Returns the number of bits in mask set to 1 out of the first
// n least significant bits.
int countBits(int mask, int n) {
     int res = 0;
     for (int i=0; i<n; i++)
          if (((mask & (1<<i)) != 0)
              res++;
     return res;
}
```

11) (8 pts) An n-digit sum divisible number is one such that for each prefix of k digits, the sum of those digits is divisible by k, for each value of k, ranging from 1 to n. For example, 372495 is a 6-digit sum divisible number because 3 is divisible by 1, 3+7 is divisible by 2, 3+7+2 is divisible by 3, 3+7+2+4 is divisible by 4, 3+7+2+4+9 is divisible by 5 and 3+7+2+4+9+5 is divisible by 6. Complete the code below so that it prints out each 7-digit sum divisible number that doesn't use repeat digits and doesn't start with a 0. (Note: we store a number digit by digit.)

```
#include <stdio.h>

void go(int digits[], int used[], int curSum, int k, int n);
void print(int digits[], int n);

int main(void) {
    int digits[7];
    int used[10];
    for (int i=0; i<10; i++) used[i] = 0;
    go(digits, used, 0, 0, 7);
    return 0;
}

void go(int digits[], int used[], int curSum, int k, int n) {

    if (k == n) {
        print(digits, n);
        return;
    }

    int start = (k > 0) ? 0 : 1;

    for (int i=start; i<10; i++) {

        if ( ____________ ) continue;

        if ( ________________________________________ ) continue;
        used[i] = 1;
        digits[k] = ____ ;
        go(digits, used, _________________  , ________ , n);
        used[i] = _____;
    }

}

void print(int digits[], int n) {
    for (int i=0; i<n; i++) printf("%d", digits[i]);
    printf("\n");
}
```

12) (8 pts) You have discovered that if you study for a final exam for ***t*** hours, the score you will obtain on the final is expressed by the function $f(t) = \sqrt{t} + 2^{\frac{t}{10}}$. You have determined that you need a score of ***desiredScore*** to obtain an A in the course. Complete the function below so that returns the number of hours (real number) you will need to study to obtain exactly the score of ***s*** (within a reasonable tolerance), so that you don't "over" or "under" study.

```
#include <math.h>

double amtToStudy(double desiredScore) {


    double low = 0, high = desiredScore*desiredScore ;

    for (int i=0; i<60; i++) {









    }

    return low;
}
```

13) (1 pt) What holiday is celebrated by many the day after Christmas Eve? _____