2016 Spring COP 3502 Exam #1 - Dynamic Memory Allocation (20 pts)

Date: 2/18/2016

Name: _____

Consider a video game with multiple levels, where each level can be represented with a two-dimensional array of integers. For the following questions you will read in information about the number of levels and the dimensions of each of the levels and allocate the appropriate amount of space to store the information.

1) (5 pts) Complete the blanks in the segment of code below to read in a single integer from standard input into the variable numLevels and then allocate that many pointers to two dimensional arrays. (You will not allocate the space for any of the 2D arrays yet, just for the pointers to those arrays.)

```
int numLevels;
```

_____ ;

```
int*** game =
```
_____ ;

2) (13 pts) Complete the blanks in the segment of code below so that it reads in numLevels pairs of integers, r_i and c_i ($0 \leq i \leq$ numLevels), where the ith pair are the dimensions of the ith level, and dynamically allocates space for each level using the appropriate set of mallocs.

```
int i, j, numRows, numCols;
for (i=0; i<numLevels; i++) {

    scanf("%d%d", &numRows, &numCols);
```

_____ = _____ ;

_____

_____

```
}
```

3) (2 pts) If the input to the segments of code above was

```
3
2 10
8 5
3 6
```

how many malloc statements would be executed by the segments of code above?

_____

4) (10 pts) Determine a closed for solution to the following summation, in terms of n:

$$\sum_{i=2n+1}^{3n} (2i - 1)$$

_____

5) (10 pts) Consider the process of solving the recurrence relation $T(n) = 4T\left(\frac{n}{2}\right) + n^3$ using the iteration technique. Carry this process out to the third iteration where you get an equation of the form $T(n) = 64T\left(\frac{n}{A}\right) + \frac{B}{C}n^3$, where A, B, and C are each integers, determining the values of A, B and C. (Partial credit will be given for finding the correct equation at the second iteration.)

6) (10 pts) An $O(n^2)$ algorithm takes 100 ms with an input size of n = 50000. Roughly, how long, in ms, will the algorithm take on an input of size n = 80000?

Date: 2/18/2016

Name: _____

7) (10 pts) The 3n+1 problem is as follows: Given a positive integer, *n*, calculate the next number in the sequence by dividing *n* by 2, if *n* is even, or calculating $3n + 1$, if *n* is odd. Continue the sequence until the number 1 is generated. If n = 7, the sequence created is as follows:

7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1

Thus, if we start with n = 7, the sequence lasts for a total of 17 terms, including 7 and 1.

Write a ***recursive*** function that takes as its input a positive integer n and returns the number of values in the sequence generated by n using the procedure described above. You may assume that the sequence terminates in 1000 steps or fewer and that none of the intermediate values generated overflow the int data type.

```
int threeNPlusOne(int n) {

    if ( _____ )

        return _____ ;

    if ( n%2 == 0)

        return _____ ;

    return _____ ;
}
```

8) (10 pts) The following function is an attempt at fast modular exponentiation. Much to the chagrin of the student who wrote it however, it seems to run equally slow as the iterative version. Explain why this code runs slower than expected. What needs to be done to speed it up?

```
long long fastModExp(long long base, long long exp, long long mod) {

    if (exp == 0)
        return 1L;

    if (exp%2 == 0)
        return (fastModExp(base,exp/2,mod)* fastModExp(base,exp/2,mod))%mod;

    return (base*fastModExp(base,exp-1,mod))%mod;
}
```

9) (10 pts) Write a ***recursive*** function that calculates the sum of the digits of its input integer n. (Note: n is guaranteed to be non-negative.)

```
int sumDigits(int n) {




}
```

Date: 2/18/2016

Name: _____

10) (12 pts) Consider a Merge Sort of the 8 elements shown below. In the process of the sort, the Merge function gets called 7 times. Show the contents of the array after each of the Merge function calls completes (except the last):

Index	0	1	2	3	4	5	6	7
Original	13	27	12	9	30	15	6	3
After 1st Merge								
After 2nd Merge								
After 3rd Merge								
After 4th Merge								
After 5th Merge								
After 6th Merge								
After 7th Merge	3	6	9	12	13	15	27	30

11) (6 pts) Show the contents of the following array after each iteration of Bubble Sort.

Index	0	1	2	3	4	5	6	7
Original	13	27	12	9	30	15	6	3
After 1st Iteration								
After 2nd Iteration								
After 3rd Iteration								
After 4th Iteration								
After 5th Iteration								
After 6th Iteration								
After 7th Iteration	3	6	9	12	13	15	27	30

12) (2 pts) The singer Adele Laurie Blue Adkins is better known by what first name?

_____