CS300 Exam2 Review

1) Write a C function that accepts an array and the number of elements in the array. Return true if there are any duplicate elements in the array; otherwise, return false.

2) What is the computing complexity of your solution in 1)?

3) Show using the definition of Big-O that your computing complexity in 2) is correct.

4) Given a positive number n, write a function to determine whether the number n is the sum of its divisors.

5) What is the computing complexity of your solution in 4)? Explain.

6) Create an ADT for the mathematical concept of a set. Your ADT is to include the operations: setCreate, setInsert, setRemove, setIsIn, setUnion, setIntersection, setDifference.

7) What is a reasonable representation in C for the set ADT described in 6)?

8) Using the representation described in 7) implement setCreate, setInsert, setRemove, and setIsIn.

9) What is the computing complexity for each of the set operations described in 6)? Explain.

10) Create an ADT for the mathematical concept of a bag. A bag is similar to a set except a bag may contain duplicates. Your ADT is to include the operations: bagCreate, bagInsert, bagRemove, bagIsIn.

11) What is a reasonable representation in C for the set ADT described in 10)?

12) Using the representation described in 11) implement bagCreate, bagInsert, bagRemove, and bagIsIn.

13) What is the computing complexity for each of the bag operations described in 10)? Explain.

12) Show that the following statements are correct using the definition of Big-O:

a) $10n^2 + n + 9$ is $O(n^2)$

b) $6n^3 / (\log_2 n + 1)$ is $O(n^3)$

13) Consider the following C declarations:

```
typedef struct
        {
          int x, y;
          float z;
          int abc[10];
        } Foo;

int values[5][10];
Foo aFoo;
Foo arrayFoo[10];
```

a) Give the general accessing formula for find an arbitrary element in values.

values[i][j] =

b) How many bytes of space are taken up by the struct aFoo?

c) If the base (arrayFoo) is 1000, what is the starting address of arrayFoo[5].abc[5]?

14) Using your list functions, how would you concatenate on list onto the end of another list? What is the computing complexity of this operation?

15) Review the following:

a) Stacks & stack representations

b) Strings & string representations

c) Pointers vs Arrays

d) Dynamic memory allocations

e) Activation Records & the heap

16) Review all notes and see me if you have questions