CS300 Exam2 Review

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

2) What is the computing complexity of your solution in 1) (best case, worst case, average case)?

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

4) What is the computing complexity of your solution in 3)? Explain.

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

6) What is a reasonable representation in C for the set ADT described in 5)?

7) Using the representation described in 6) implement setCreate, setInsert, setRemove, and setIsIn.

8) 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? How can you write C code to check your answer?

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

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

10) Review the following:
a) Stacks & stack representations
b) Strings & string representations
d) Subversion
e) Come up with an example of a usage of extern and an example for static. Describe the errors that occur if extern and static, respectively, are removed from your examples.

f) Explain how to use typedef to create a datatype.
g) review the coding standard variable name prefixes.
h) review the Linux command line
I) review all your makefiles.  How can you use diff to help you test your code from the makefile?

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