Sample Exam Questions

This document contains examples of the types of questions that will be on the first exam, not necessarily the content of those questions. In other words, its purpose is to illustrate the format and style of the questions. Some questions may be more difficult than those presented here.

1. What are the components of a compiler collection and what do they do?
2. Explain what is meant by data abstraction.
3. Write a recursive definition of the function $f(n)$ that computes the n^{th} Fibonacci number.
4. What value is returned by the call $f(3)$ where f is the recursive function defined by the following code?

```
int f( int n )
{
    if ( 0 <= n )
        return 0;
    else
        return f(n-1) + n;
}
```

5. Given the following constructor definition in the class `myClass`,

```
class myClass  {
public:
    myClass ( type1 x, type2 y = value2, type3 z = value3) {}
    ...
};
```

show all the ways in which an object of `myClass` could be declared.

6. Write a function to compute the sequence of disk moves for the Towers of Hanoi problem.
7. Write an implementation of the following member function of the `List` class, using either an array or a vector private member named `Items` and with current size `Size`. You can assume the array has room in it.

```
void List::Insert(int Position, ListItemType NewItem, bool& Success) ;
// Precondition: 1 <= Position <= Size +1
// Postcondition: Inserts an item into position Position
```

8. For each of the following statements, circle **T** if it is correct and **F** otherwise.

   (a) **T F** The parameters that should not be modified by a function should be passed to it by non-constant reference.

   (b) **T F** All objects declared in a program are destroyed only at the time of program termination.

   (c) **T F** The base case is only necessary to make recursive functions run faster.

   (d) **T F** An iterative solution, if it can be written clearly and cleanly, is more efficient then recursion because it avoids function call overhead.