CSI 1440 **Review 2**
Summer 2005

1. Create a polymorphic type with three different kinds of things, and implement a function called “Process” that processes each element of the type. Thing A has two integers that need to be added together to form a third integer, Thing B has one float that needs to be squared, Thing C has a character string and an integer, you must find the length of the string and store it in the integer.
2. What is the purpose of multiple inheritance (usually)?
3. Create a function that finds the midpoint of its two arguments. This is done using the formula (a+b)/2. This function must work for any type that defines addition and division.
4. Create a function called “scale” that takes three integer arguments, p, n, and d and produces the result (p*n)/d. The function should throw an exception of type “ReallyBad” if d=0. Define “ReallyBad” as an exception class.
5. Create a class that contains an array of some type with three functions. The class must have no default constructor, and a constructor that has one integer argument giving the size of the array. The other two functions are “Some Type Get(int ix)” and “void Set(int ix)” These functions throw an exception if ix is invalid.
6. Create a header class and an item class for a linked list that contains one char *. Create appropriate constructors and destructors for each. Create an Add function for the header class that is defined void Add(char * NewChar).
7. Create a header class for a stack. The items should contain two integers. Implement the push and pop operations.
8. Define the Node class for a binary search tree containing floats. Create an Add function to add a new node and a print function to print the vertices in ascending order.