CSE 373 – Data Structures
Homework 1

Assigned:	Wednesday, April 3, 2002
Due:	Wednesday, April 10, 2002
	At the start of class

Remember to also do an electronic turnin of list.c.

Your name: ____________________________

Student number: ____________________________

1. Consider the function **`sum`** that you compiled and ran as part of the homework 1 project. The first parameter of the function is defined as ElementType v[], an array of pointers to objects. The third parameter gv is also a pointer. What does it point to?

2. The example in the Sum project description (step 5) shows the result of running mainSum with the file sumsymbols.txt, a small symbol table that defines 5 symbols.

   Draw a picture similar to slide 15 in the lecture of April 3 that shows each of the entries in the array ElementType v[] when **`sum`** is called from mainSum at line 119. Also show each of the objects that v[] has pointers to, and each of the objects that they point to, and so on. Show the values (1,2,3,4 and 5) and the names (“one”, “two”, “three”, “four” and “five”) in the proper locations in the drawing. You can indicate the pointer values by drawing arrows, you don’t need to supply actual addresses.

3. Consider the function CreateList that you wrote as part of the List implementation. The function returns a value of type List which is a pointer. The program mainList calls CreateList and sets the value of variable L using the returned value at line 76. Draw a little diagram that shows variable L, the object that it is pointing to, and the contents of that object, after line 76 is completed.

4. The examples in the List project description include a run of the program with the file "twosymbols.txt", a very small symbol table that defines only two symbols.

   Draw a picture similar to slide 18 in the lecture of April 5 that shows the list pointer L, the object it points to, and all other objects that are linked together at the moment when mainList is about to call PrintList on line 94 during the run with twosymbols.txt. As in question 2, show the values (1 and 2) and the names ("symOne" and "symTwo") in the proper locations in the drawing. You can indicate the pointer values by drawing arrows, you don't need to supply actual addresses.

5. Consider the function CountListEntries that you wrote as part of the List implementation. Does the number of operations that this function performs vary depending on the number of entries in the list? Why or why not?