CS 2401 – Fall 2021

Project 5 – Inheritance and Virtual Functions

Due: 11:59 PM Friday, November 12th

A lot of different things end up being stored in a closet, and the attributes that would be used to describe those things vary greatly. You would not use the same variables (or attributes) to describe a sweater that you would use to describe a pair of pants, let alone a soccer ball or that childhood teddy bear that sits in the back corner.

For this project you are to create a group of classes that describe things that might be in one's closet. Since they have the common feature of being stored in the closet (a lot of them might be clothes, but as I just pointed out there are other things in there as well), there will be a parent class called something like "Closet_item" and then a minimum of five other classes which are children of this class. We're not going to be doing any real calculations using these items, so all they really need are input and output functions, which will need to be virtual in the parent.

You are also to construct, for 10% of your grade, a UML class diagram which describes your classes, but does not need to describe the application. This diagram is to be constructed using Star UML and should be in the form of a .mdj file, that is included with your project submission. (Star UML is available for free at https://staruml.io/download)

The application for this project will declare an STL list of pointers of the parent class (Closet_item) which will hold dynamically allocated closet items. The user will be presented with a menu from which they can choose a closet item that they would like to add to the closet. They will then be taken to an input function for that item where they can enter its attributes. Once the customized item is constructed it is put into the list with the push_back member function.

The menu should also offer a means to view all the items in the list. Since the list class has no "show_all" function, we will have to rely on the iterator to do this, and note that since we are storing pointers to items instead of the items itself, and since non-member overloaded operators don't work well with this paradigm, the syntax for outputting each thing will look like: `(*it)->output(cout);`

Finally, you need to write a backup file system so that a user does not have to reenter everything in their closet every time they start the program. This backup file is to be invisible to the user. The program simply looks for the file, and if it is not found it is assumed that this is the first use of the program by this user, but if it is found the items are put into the list the same way as they were when entered from the keyboard. Following the loading of the list from a file the user is taken to the menu where they can see the stuff that is already listed as being in their closet and add more stuff to the closet. Nothing is ever removed from the closet – which is why it is so full.

Please note: You are NOT writing a container but are instead using the STL list class. (Available if you just #include <list>.) For this project you will just need the push_back function and the iterator, which is bi-directional, but which you will only be using in a forward direction for closet output.

Also note, I am giving you no code for this project – you are to write it all yourself, and variety and creativity of the classes is a factor in determining your grade. You are to write a total of six classes and the main which will include the menu that works for your distinct set of classes.