

Programming Assignment #1

CS 202 Programming Systems

***** Make sure to read the Background Information first!**
It applies to all programming assignments this term***

Always backup your files PRIOR to using tar
Double check that the arguments specified with for tar are correct!
Penalties will apply to all submissions incorrectly archived/uploaded

Background:

When beginning with this project, the first thing to keep in mind is that we are no longer working on CS163 programs! In CS163 we were concerned about creating Abstract Data Types and the class construct facilitated this. Instead, this term we will be focusing on how to create Object Oriented Solutions. An ADT may be part of that solution – but it certainly shouldn't be the primary focus. Instead you want to strive for classes to have specific “jobs” and have classes derived from more general classes, whenever appropriate. We will be working in situations where there are multiple classes, so you will want to focus on dividing the design into smaller components that have specific jobs working together to solve the problem.

Every assignment this term needs to have at least 5 classes. With these, think about how to design the classes such that they reduce the amount of work another class needs to do. The idea is if we have “robot” like classes doing the smaller tasks or “jobs”, that by the time we get to a larger class that has more to manage – it will have little left to do! We can achieve this by delegating. Often the over-use of “getters” can cause the opposite to happen – and instead of delegating the managing class has to fundamentally do all of the work itself.

**Lab #1 and Lecture #2 will cover the concepts of
building object oriented programs!**

Overview of a Concept:

Did you watch the news over the break? There were so many traffic jams around Mount Hood. My family was stuck on the back side of the mountain for three hours because of over crowding; everyone wanted to be on the mountain at the same time even during a snow storm! Amazingly, people were not prepared to drive in winter conditions. In some cases, cars were diverted around the mountain to Hood River to get back to Portland. This is not a simple drive. There seems to be a disconnect in getting the right information to skiers and drivers before they start their journey up the mountain. Trip check is a good start, but it is not predictive. Of course, our two lane mountain roads were not designed for the population growth we have experienced here in Portland. Not everyone can be on the mountain on a nice day. In fact, if you can make it up the mountain, it is even more disappointment to find out that the parking lots are all full and that it is impossible to ski that day after all.

Specifics about Program #1

Your job for program #1 will be to develop software that can predict how busy the roads will be for skiers and those that want to play in the snow so that they can make decisions before leaving Portland and driving to Mount Hood specifically. Drivers will want to know the best route to take, the number of hours it will approximately take them to get there, how long it will take to get home when they are done and if special traction devices will be needed.

Some of the information you will need to keep track of includes:

1. The roads (Highway 26 or Highway 35/84)
 - a. Current volume on the roads
 - b. The road conditions (bare pavement, icy, etc.)
 - c. Where there are accidents
2. How weather will impact the trip
 - a. On a nice day there will be more cars
 - b. On a snowy day it will take longer to chain up and cars will head up the mountain that do not have the proper traction devices causing all sorts of problems

Then, select ONE other characteristic to consider in your software:

1. Number of skiers expected at the resorts
2. How full the parking lots at the ski resorts are (including Mt Hood Meadows, Timberline, Ski Bowl, Summit, Cooper Spur, and Snow Bunny)
3. Changes expected in the weather as the day progresses
4. Others?

Once you make a decision about what aspect you want to support - you will want to think about breaking this down into a series of classes and create them independent of the entire problem. Some relationships should be hierarchical, others can be containment. With hierarchies always push the common elements up to the base class. Avoid classes with only setters and getters! And Nodes will need to be classes instead of structs.

Required Data structures

This program should implement the following data structures. All traversal functions must be implemented **recursively**.

1. An array of linear linked lists for the highways surrounding Mt Hood which includes the information about traffic volume and accident location
2. Implement another data structure (of your choice) that represents the user's journey and the factors that will influence their travel to the mountain and back down when they are done.

Brainstorming (The following are suggestions)

The following represents some ideas on the design – the first step is to plan what classes might make the most sense! Some of the basics that are part of an OO program **could** be:

1. Highways
 - a. Contains the mile markers where accidents commonly take place and the current status of ice or snow at these points.
 - b. Contains information as to the amount of delay expected at each of these mile markers
2. Journey
 - a. My path from home to and from Mt Hood and all of the items that impact that journey
3. Weather
 - a. Weather that impacts our journey includes if it is sunny, icy, freezing rain, windy or snow-fall and the time impact that such weather influences. A light snow is different than a blizzard. If these are handled by a class, the rest of the code will not need to worry about the details of how weather impacts them.

Anything that is similar between these or other classes that you write should be pushed up to be part of a base class. For example, classes that manage collections of items may be derived from a common base class that manages the collection. Keep classes small and functions small. A large class or function means that the problem has not yet been broken down into its basic components (objects).