Assignment 1: Design

January 28 2018

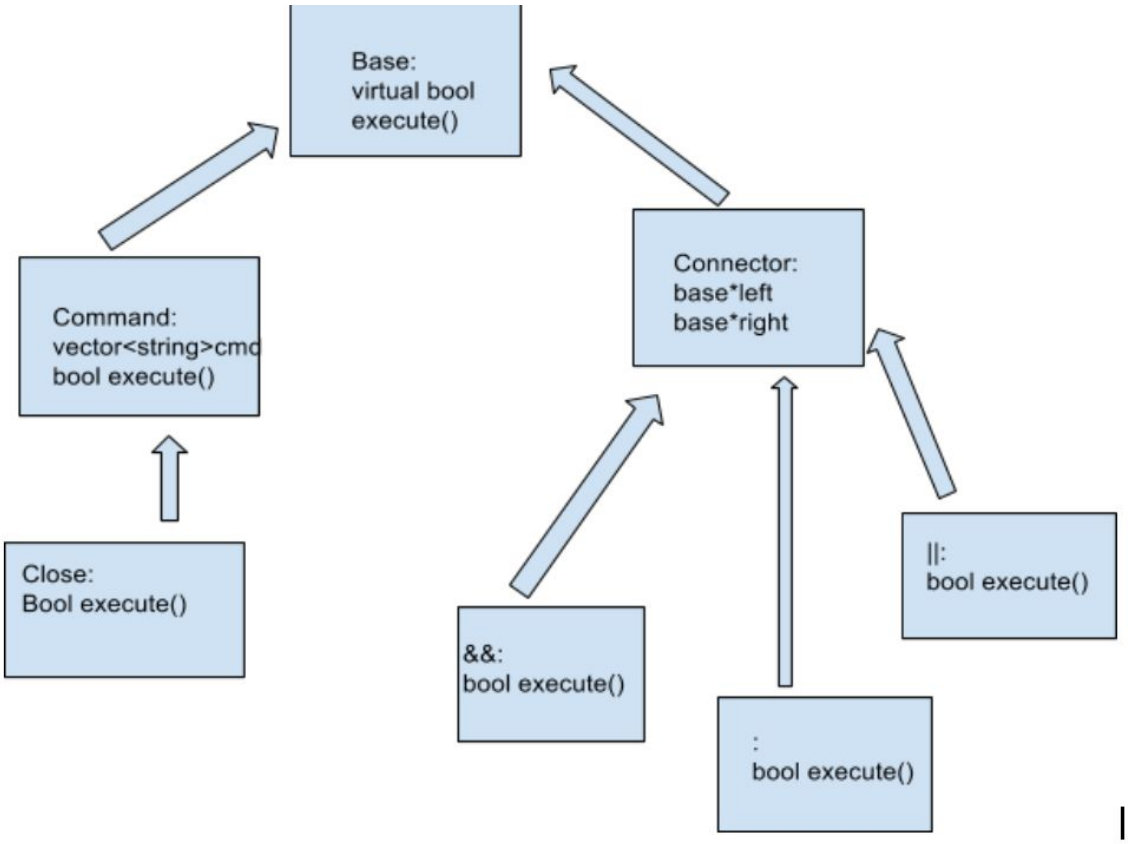
Winter 2018

Katherine Way and Keyu Wang

Introduction:

For this program, we will be using C++ to create a command shell known as “rshell”, which will be able to read in and process various terminal commands that the user inputs. We will use a vector of strings to store the user’s input, and then store these input lines into a stack for processing. This program should be just like a simulated terminal, and as such will print out a command prompt and read in the user’s input from one line.

Diagram:



Classes/Class Groups:

Base

This is our abstract base class for the project, and will contain a virtual function bool execute() that will be extended by other classes.

Connector

Connector is a composite. It has two pointer members which will be used in the construction of its children.

&& (and)

&& is a leaf which inherits from Connector. It will first execute left, and will then execute right if and only if left is run successfully.

;

; is a leaf which inherits from Connector. It extends the execute function by running both left and right, regardless of the result.

|| (or)

|| is a leaf which inherits from Connector. It will execute right if left is unsuccessful.

Cmd

Cmd is a vector of strings that will store the user inputs. This class will also extend the execute function from the base class by using the strings inside of the cmd vector.

Close

Close is a leaf that inherits from cmd. It will close the program by returning a value of -1.

Coding Strategy:

For this project, we will use the “Driver - Navigator” style of pair programming where Keyu will be the Driver and Katherine will be the Navigator. If there is a feature that one of us have more knowledge than the other, the positions will be switched.

This is to avoid having conflicting directions of the project and help progress the development of the project rather than trying to fight for control of the direction of the project.

Roadblocks:

During the development of this project, the following issues may arise:

Schedule Conflicts

My partner and I may have schedule conflicts and therefore will not be able to always meet up and talk about the progress of our project. In addition, it may affect the time we can commit to polishing the project because of not being able to meet up.

Incorrect Implementation of Code

There may be times where my partner and I incorrectly implement the code due to lack of knowledge of efficient coding. Overestimation of our skills may hinder on the progress we plan to make during a time period. Because of this, progress could be stopped as we continue on with the project. Functions may not work, unfamiliarity of certain coding methods but this can be avoided talking and seeking out help from resources such as professor and TA.

Overestimation of Time

Because there may be times where unexpected events come up, the time we plan to finish certain features may end up being off. This may cause certain features to not be finished or have sloppy code implementation.