Assignment 1: Design

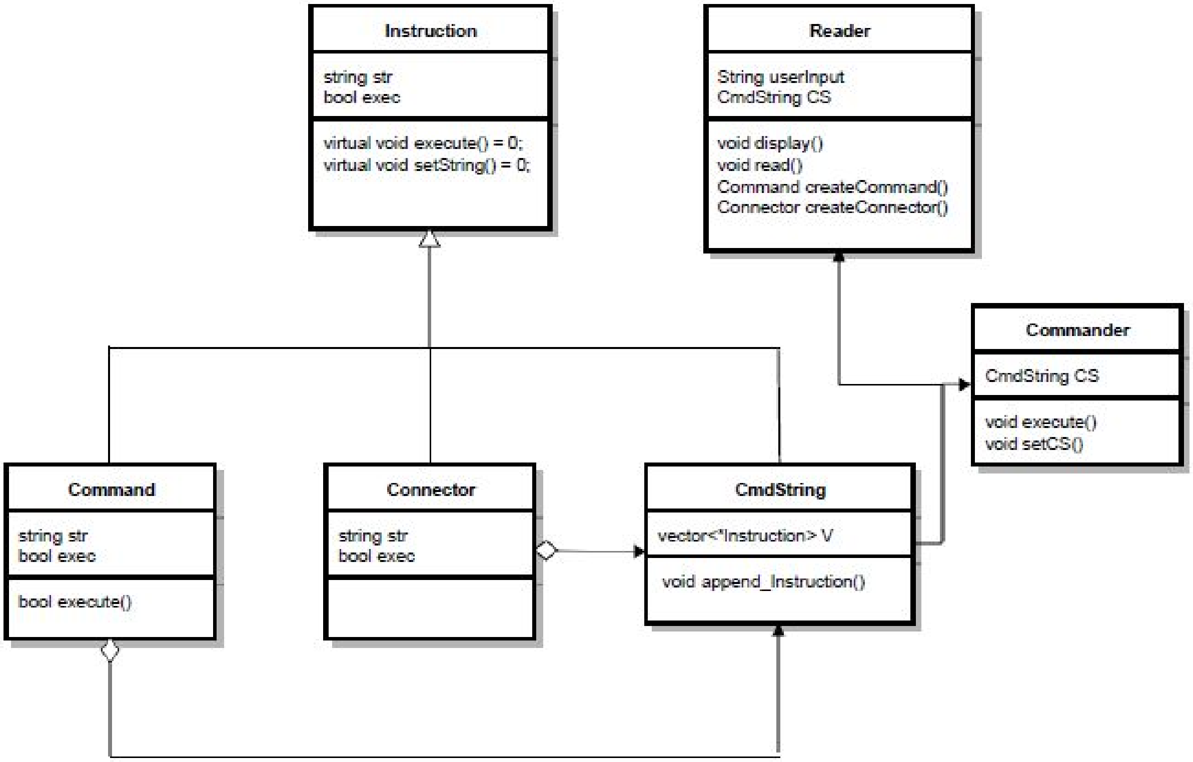
10/20/2017

Fall 2017

Kyle Banda
Clement Veaux

Introduction:

In this assignment we are designing a command shell called RShell. Our objective is to be able to input a command string such as “ls; cat main.cc && main.h”, and have our RShell run each command in sequence if the connectors (;, ||, &&) allow it. Our design has a composite strategy which allows us to break up a command string into commands, connectors, and smaller command strings. The Reader class will read user input and facilitate the creation of the previously mentioned commands, connectors, and command strings as objects. It will then pass it on to the Commander class which will carry out commands and compare their boolean values with connectors. After making each comparison, the Commander decides what to do with the next command in the sequence.



Classes:

- Group: Instruction
    - Instruction
        - Base class for Command, Connector and CmdString classes - has a string str, bool exec
    - Command:
        - Inherits from Instruction
        - Has function execute() which carries out the command in its string object
    - Connector
        - Inherits from Instruction
    - CmdString
        - Inherits from Instruction
        - Has vector<*Instruction> so that it can hold Commands and Connectors
        - This class will be passed to Commander so that it can execute the function
- Reader
    - Reads user input and parses it
    - Separates user input into Commands and Connectors and puts them into a CmdString
    - Passes the CmdString onto Commander
- Commander

- Executes commands and checks them with the next connector. It will or will not execute the next command based on the bool comparison between the Command and Connector

Coding Strategy:

We will split up the design into two parts. We will work separately on the Instruction inheritance group and the Reader and Commander classes.

Roadblocks:

In this assignment depending on how we code our classes we may have to change strings into char arrays/vectors depending on what compatibility requirements there are. One example would be strtok() which accepts a char pointer to a char array. We would either have to convert from string to char arrays or use char arrays exclusively to make coding easier for ourselves in the future.