User Manual

Program: Homework 4.cpp

This program prompts you for whether or not you want to receive instructions on how to play the game. It then asks you which peg you would like to remove to begin the game. The game then allows you to choose a slot peg to jump over, and then the peg with which you would like to jump. You repeat this process until you either run out or there is only one peg remaining. If there’s a single peg, it will give a win message, if there are no moves, you must exit using the exit arrow at the top right.

To run the program double click on the file Homework 4.exe, or open Homework 4.cpp with Microsoft Visual Studio 2010 or above and hit Control F5. A console window will appear asking you to enter the y or n for yes or no for instructions. It will then ask you for coordinates in the form “B5” repeatedly. Hit enter after typing your input for each field. Upon victory it will print “YOU WON!!!!! CONGRATULATIONS!!!" Press any key to terminate the program when you are done.

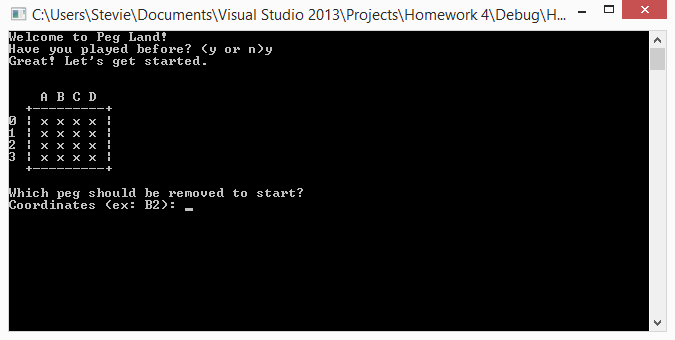
System Manual

Program: Homework 4.cpp

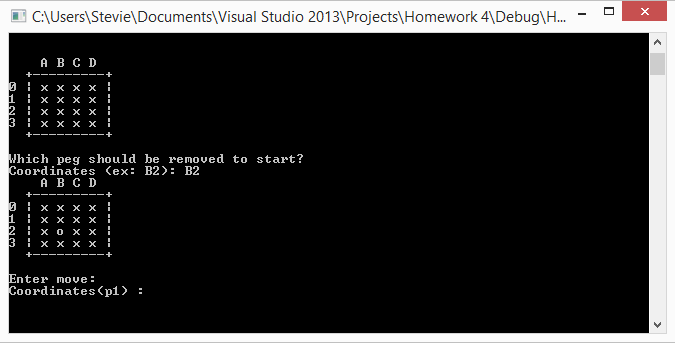
This program focuses on creating a peg game. It uses the information gained from the user to get locations for movement. The user enters the names of private variables, which are turned into getters and setters as methods. The program runs through stages of establishing data, the initial peg movement, and then the moves following. It also uses two functions, one to translate a character to a number and another to prompt the user for information regarding moving the pieces location.

The user is expected not to use special characters that C++ will not recognize. They are also expected to enter a non-negative integer for the second location. The input is declared as a char and an integer. The program malfunctions if you try to use anything but a number using 0-9 for the first input or anything but a char for the second it skips past all other commands…

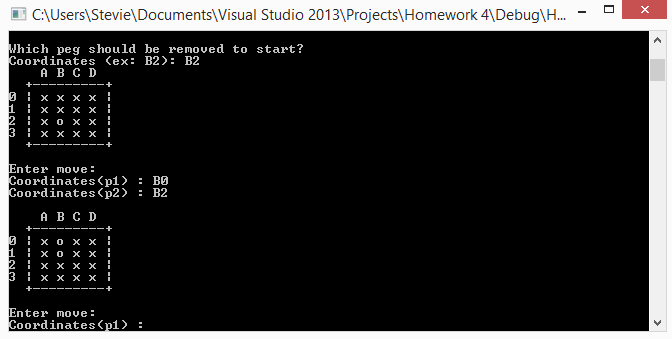
Test Log



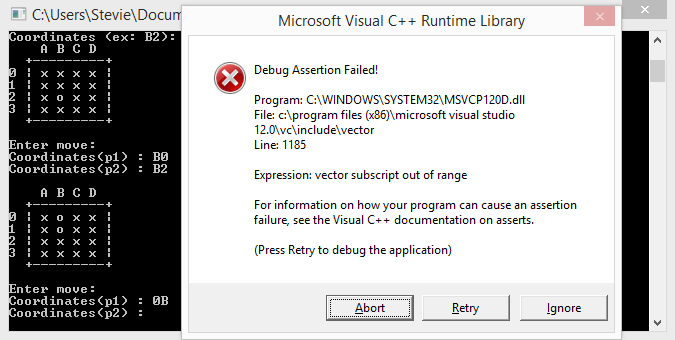
Example of instructions



Example of initial peg removal



Example of jumping pegs (order of points doesn’t matter), also example of user input.



Trying to enter invalid input