8: A Graphic Interface

CSCI 4526 / 6626 Fall 2016

1 Goals

- To experience using Unicode to make the kind of graphics display we used to use before everyone had color graphics screens.
- To integrate a text-based graphics viewer, written by someone else, into the application.
- To use an abstract class.
- To achieve a working application that can play a *Game* of sudoku with you.

2 The Viewer Interface

As part of this project, I am supplying a Unicode-character graphics implementation of a Sudoku-*Board* viewer. This viewer is built from several classes – you need to download all of them and put them (verbatim) into your project. Do not change anything. Then just compile and run normally. The classes are:

- CanView: an abstract class. Include its header file in your *Board*.hpp.
- Viewer: renders the sudoku *Board* in a readable manner. Include Viewer.hpp in your *Game*.hpp.
- Canvas, GridChar, and MixedChar: three classes that are used by Viewer. Include both code and header files in your project. You do not need to do anything else about these classes or with these classes. You can use them without understanding them, or look inside to learn about Unicode-8.

Using the Viewer. Derive your Board class from the abstract class CanView. In your *Game* class, you must first instantiate the *Board*, then instantiate the Viewer, giving it a reference to the *Board*: `Viewer fancyView(9, 9, myBoard)`. To display the *Board*, call `fancyView.show(cout);`

Trouble. Please report right away if you have any of these kinds of trouble.

- You can't compile your project with this package included.
- The screen does not show a nice-looking Sudoku board. Be sure your screen font size is small enough to show a whole row on one line.
- You have trouble getting the Unicode to work at all. In Linux, you have to tell your shell that you are using Unicode.

3 Testing and Submission

Try using the menu to make some moves on the *Board* that you constructed from the original input file. Test both meaningful and illegal moves.

This is due November 7. Please submit screen shots of the moves you make and a screen shot of the resulting *Board*.