Projects Portfolio

Overview

You are to create a report that presents a portfolio of all of your work over the semester.

For each assignment, the portfolio should include:

1. The first page with your name and list of assignments.
2. The PS number and title of the assignment.
3. A discussion of the assignment itself and what you accomplished.
4. You should have the separate discussion for every part of assignment.
5. A discussion of one or more key algorithms, data structures, or OO designs that were central to the assignment.
6. A discussion of what you learned in the particular assignment.
7. Evidence that the code ran -- e.g., screenshot, console log, text output, results of running tests.
8. If you didn't complete any part of any assignment, the things that you didn't do, or didn't get working.
9. Nicely-formatted source code listings for all source code comprising the assignment:
    - **Code must be displayed in a monospaced font**
    - **Should display line numbers**

Please remember that this assignment is worth 10% of your overall class grade.

The assignment is due at 11:59 pm on December 6. Late work will not be accepted.

Format

The document must be a delivered as a single PDF file.

The document should have the following sections:

- *Cover page and table of contents.* This should be a single page that has your name on it, the date, and a listing of each assignment.
  See pp_cover_Fall2019.docx as an *example*.
- For each assignment, the items numbered 1 through 7 above, *in the order they were assigned.*

Additional notes:

- Make sure to include a discussion of what you learned in the assignment. Explain what you already knew and where you learned it. On the other hand, if some aspects of the assignment challenged you or you what to learn more, describe this.
- Code listings should be provided in this order:
  - `Makefile`
  - `.cpp` source for the main routine
  - `.hpp` source for supporting class file
  - `.cpp` source for supporting class file
  - Any additional `.hpp` and `.cpp` files as part of the assignment

Do NOT put all the code at the end of the document. Instead, put the files associated with each problem set immediately after the narrative for that PS.

Document Production Tools

You may use any tools you like to produce the document. The rest of this section presents a set of open-source document creation tools you may wish to use:

- `enscript` — creates nicely-formatted PostScript output from text source code files.
- `ps2pdf` — converts PostScript files to PDF files.
- `pdftk` — "PDF Tool Kit" — will assemble multiple PDFs into a single PDF.

enscript

The Unix utility `enscript` may be used to make formatted versions of your source code files. The `-C` option adds line numbers (which you can then use in your narrative to refer to specific sections). I like the `--margins=50:50:50:50` to set up the margins nicely. Use the `-ooption` to direct the output to a file.

This will produce a PostScript (`ps`) output file. Then, use `ps2pdf` to make a PDF file.

E.g., for the `GuitarHeroLite.cpp` starter file, use these commands:

```
enscript -C --margins=50:50:50:50 GuitarHeroLite.cpp -o
GuitarHeroLite.cpp.ps
ps2pdf GuitarHeroLite.cpp.ps
```

and then you will have a file named `GuitarHeroLite.cpp.pdf`.

A word processor that outputs PDFs

Use any word processor you like and save individual assignment narratives as PDF files.

pdftk

Use the Unix utility `pdftk` to assemble separate PDFs into a single one.

If you name your files like this:`00_narrative.pdf`
`01_code1.pdf`
`02_code2.pdf`
Then you can assemble these using:

```
pdftk ??_*.pdf cat output section_name.pdf
```

Create numbered sections for each assignment, then assemble them into the whole PDF.

Turn in

Turn in complete portfolio as a single PDF file.

Name the PDF file `Lastname_Firstname_Comp4Fall2020.pdf`.

Submit to: **Blackboard**