**Concepts of Operating System**

**Project: Portable File Manipulator (pofm)**

**Note:** this project will be done in groups of 3 or 4 people.

Deadline for the demo: December 03, 2020

Deadline for the report: December 10, 2020

**Details**

Design a portable file manipulator, called *pofm*, which will operate on files. *pofm* must have the following attributes:

* Since it must be portable, it must NOT use any OS commands which means that *pofm* can be easily ported to any operating system and platform. For example, if you use C programming, all commands must be standard C functions, and you cannot use the system() calls.
* *pofm* must provide the following functionality:
  + Create a new file
  + Delete a file
  + Rename a file
  + Copy a file
  + Move a file from one directory to another
  + For text files only:
    - Append text to the end of a file
    - Insert text in a specific position (counted in characters) of the file
    - Remove all text in a file
    - Show the content of a text file, with the ability to pause per page. The number of lines per page can be specified by the user.
* All commands must have a “help” utility; i.e., if the user types command1 /h, for example, then the program must show how the command is used.
* All commands must catch exceptions and problems; e.g., file does not exist, file is read-only, file name already exists, invalid command usage, etc.
* You can name your commands as you like.

**Deliverables and Marking Scheme:**

1. In-class **demo** **(40%)** no later than December 03., in the last week of classes. The GA will come to your groups, one by one, you will show your *pofm* running, and the GA will ask you to show specific features and also will ask you questions about your system and about your code.
2. **Final report (60%)**: no later than December 10. This report consists of 3 components:
3. A design guide (30%): start by a short introduction, followed by explanation of your design and implementation. Give reasons for your design and implementation choices. There should be enough details in your report to explain everything. I do not set number of pages: it’s up to you and how many pages you need to explain things.
4. A user manual including examples and screenshots (20%).
5. All your code, with in-line comments (10%).