

COP 4600 - Homework 5

Due April 11, 2022

Total points: 4 + 1 bonus points

This homework requires the extension of the shell program from HW3 with several new, file system related commands. The program you are submitting must have all the functionality of HW3 as well.

Compilation: The objective of this homework is for you to learn how to implement functionality with the basic APIs. This is not about how to find some libraries that perform these tasks for you. Your code must compile out of the box on a recent Ubuntu system by simply typing:

```
gcc mysh.c
```

or

```
g++ mysh.cpp
```

No external libraries are acceptable, no other configuration parameters are acceptable.

Running: The program must run without requiring the creation of additional files. Please try it out if you can run it in an empty directory.

If the program does not compile out of the box on the TAs machines, the maximum points accessible is 2.

File references

The new commands take arguments that point to files or directories. You should implement both absolute and relative paths. Absolute paths start with "/". The relative paths should be relative to the current directory in which you started the project.

Let us assume you started your program in: /home/john, and there is a directory /home/john/other. If you say

```
maik other/file.txt
```

the file will be created in the directory /home/john/other

New built-in commands

Your shell should implement the following new commands:

```
# dwelt file
```

If a regular file exists with that name, it should print
Dwelt indeed.

If the file is a directory, it should print:
Abode is.

If there is no file or directory with this name, it should print:
Dwelt not.

```
# maik file
```

This command creates an empty file and writes the word “Draft” into it. If the file already exists, it should print an error.

```
# ccopy from-file to-file
```

Copy from-file into to-file. Print an error if the source file does not exist, if the destination file exists, or if the destination file’s directory does not exist.

Extra credit (1 point)

```
# ccopyabode source-dir target-dir
```

Copy the directory source-dir and all its subdirectories, as a subdirectory of target-dir.

What to submit:

- The code as a single .c or .cpp file.
- If you implemented the extra credit part: a text file describing the syntax of the implementation, and example of use.