

Instructions

Complete the problems assigned below and turn in your answers by the due time above. To receive full credit, you must show both the relevant command(s) and output. If a question asks about what a command does, or what the output would be, give both the command and the output. Demonstrate that you actually performed the command, and didn't just guess.

Requirements - All of your shell programs henceforth must:

- a. use the shell template
- b. check for invalid arguments and command line syntax
- c. output sensible *usage* or syntax messages
- d. be appropriately and sufficiently commented
- e. define and exit with appropriate exit status codes
- f. use variables for strings used more than once
- g. not hard-code the program name

Homework Problems

1. Modify your latest version of your **rename** program (your may start with the one provided in the homework solutions). The program takes a list of paths, and renames each according to the options specified. The following specify the supported options and the additional requirements:
 - a. Options supported (can be given on the command line in any order; option-arguments must follow their corresponding option – see **intro(1)** and **getopts(1)** if desired)
 - i. `-f` force overwrite [mutually exclusive with `-i`]
 - ii. `-i` confirm overwrite [mutually exclusive with `-f`]
 - iii. `-v` verbose output
 - iv. `-c` capitalizes filenames (i.e. File1, Test, Hw2) [`-c`, `-l` and `-u` are mutually exclusive]
 - v. `-l` makes filenames all lowercase [`-c`, `-l` and `-u` are mutually exclusive]
 - vi. `-u` makes filenames all uppercase [`-c`, `-l` and `-u` are mutually exclusive]
 - vii. `-w` removes whitespace from each filename
 - viii. `-n num` numbers files; appends the integers `num`, `num+1`, `num+2`, ... to each consecutive filename
 - ix. `-p prefix` adds the prefix `prefix` to each filename [`-p` and `-P` are mutually exclusive]
 - x. `-P prefix` removes the prefix `prefix`, which may be a regular expression, from each filename [`-p` and `-P` are mutually exclusive]
 - xi. `-s suffix` adds the suffix `suffix` to each filename [`-s` and `-S` are mutually exclusive]
 - xii. `-S suffix` removes the suffix `suffix`, which may be a regular expression, from each filename [`-s` and `-S` are mutually exclusive]
 - xiii. `-r` **Extra Credit:** recursively applies changes to files under any directories given as arguments. Normally, directories specified on the command line are themselves renamed and not traversed. With the `-r` option, renames all non-directory items under the specified directory tree (this includes traversing into its sub-directories). Dot files are ignored.
 - b. Output an error and usage message for conflicting mutually exclusive arguments (i.e. `-s` and `-S`, `-i` and `-f`, etc.)
 - c. Do *not* call your program recursively – handle these requirements with only one call to your program.

2. Create a program named **dom2dir** which takes reads a list of domain names (i.e. www.yahoo.com) and creates a sub-directory tree based on the (dot separated) components of the reversed domain name.
 - a. The rightmost components of the domain name are created highest in the directory sub-directory tree, and the left-most components are created deepest in the sub-directory tree (e.g. for www.yahoo.com, the directory tree `com/yahoo/www` would be the created). Don't assume only three levels deep – any number should be handled.
 - b. Use the directory named **./TLD** as the base directory for the directory tree.
 - c. This problem *must* be solved entirely with Bourne built-in shell commands, with the exception of the external command **mkdir**.
 - d. The list of domain names contains one domain name per line (newline separated)
 - e. The program must be able to read the list as an argument, from a pipe, or from redirection. Examples:
 - i. `dom2dir urlfile`
 - ii. `cat urlfile | dom2dir`
 - iii. `dom2dir < urlfile`

Hints

- Think about how you will parse each domain name and how then to reverse the components
- The bulk of this program can be implemented using about 13 lines of code