OS A1 Documentation

Author: Arman Rajesh Ganjoo (2021018)
Project name: armsh (Short for Arman Shell)

Starting

Run the shell using the command given below:

```
$ make && ./unix-shell
```

Note: There is more documentation in the specific source files of each command (including the internal commands). Refer to them for more in-depth information on each command.

General Assumptions

- Options in a command cannot be coupled together. For example, the user cannot use the option -rf. The user must use only one option at a time.

- Some basic extra commands are provided for general use like clear, exit

Commands

echo

This command allows the user to output any message they want via the command-line.
Format: $ echo [-n/–help] <message>

Options

1. -n: Does not end the message string with a newline character (“\n”)
2. –help: Prints the help menu for the command

Corner Cases

1. Handle the case where no arguments are given
2. Not print quotation marks when printing the message
3. Handle the case where the user doesn’t close quotations

Assumptions

1. Multiline echo is not considered
2. Echo will not print any type of quotation mark if it is in a pair

Testcases

- ➔ echo hello world
- ➔ echo “hello world”
- ➔ echo -n “yes ok”
- ➔ echo –help
- ➔ echo (Corner case example 1)
- ➔ echo “hello ok (Corner case example 3)

cd

This command allows the user to change their current working directory to another directory using the command-line
Format: $ cd [-P/-L] <directory>

Options

1. -P: Handle the operand dot-dot physically; symbolic link components shall be resolved before dot-dot components are processed
2. -L: Handle the operand dot-dot logically; symbolic link components shall not be resolved before dot-dot components are processed

Corner Cases

1. Handle the case where the user tries to cd into a non-existent directory
2. User gets cd'd into their working directory if no argument is provided

Assumptions

1. No symbolic links are present in the directory

Testcases

- ➔ cd cmdbin
- ➔ cd ..
- ➔ cd -L/-P cmdbin
- ➔ cd (Corner case example 2)
- ➔ cd hello1234 (Corner case example 1)

pwd

This command allows the user to view (print) their current working directory using the command-line.
Format: $ pwd [-P/-L]

Options

1. -P: Avoid all symbolic links
2. -L: Logical pwd. Does include symbolic links

Corner Cases

1. Handle case where too many arguments are provided
2. Handle case where the current working directory is deleted (from external influence) but the user is still present in the directory

Assumptions

1. None

Testcases

- ➔ pwd
- ➔ pwd -P/-L
- ➔ pwd hello (Corner case 1)
- ➔ (Corner case 2 will most require a succinct working demo)

ls

This command allows the user to view the files and directories present in the current working directory of the user.
Format: $ ls [-a/-l] <optional:directory>

Options

1. -a: Display all the files including hidden dot files

2. -l: Display all files (excluding hidden files) line by line

Corner Cases

1. Allow usage of ls on directories relative to the current working directory
2. Handle the case where the ls'd directory does not exist

Assumptions

1. None

Testcases

- ls
- ls -a
- ls -l
- ls cmdbin (Corner case 1)
- ls ..
- ls non-existent-dir (Corner case 2)

cat

This command allows the user to concatenate files (single or multiple) of any type and view the concatenated output in the stdout.
Format: $ cat [-n/-b] <file> <optional:...files>

Options

1. -n: Number all the output lines starting at 1
2. -b: Number the non-blank output lines starting at 1

Corner Cases

1. Handle the case where the user uses cat on a directory
2. Handle the case where multiple files are provided as arguments
3. Handle invalid file names as arguments
4. Handle the case where no argument is provided

Assumptions

1. None

Testcases

- ➔ cat Makefile
- ➔ cat -n Makefile
- ➔ cat -b Makefile
- ➔ cat cmdbin (Corner case 1)
- ➔ cat Makefile test.txt acd.txt (Corner case 2 & 3)
- ➔ cat (Corner case 4)

date

This command allows the user to the current date of their System in multiple formats based on the option. The user can also see the last modification date of a file using the date -r command.
Format: $ date -r <file>
Format: $ date [-u/-R]

Options

1. -u: Print time in UTC format
2. -r: Print the time of a file’s last modification
3. -R: Print time in RFC-5322 format (present in POSIX-compliant systems)

Corner Cases

1. Handle case of invalid use of command
2. Handle case where date -r is used on a file which does not exist
3. Handle case where date -r is used without any arguments

Assumptions

1. None

Testcases

- ➔ date
- ➔ date -u
- ➔ date -r Makefile
- ➔ date -R

➔ date now (Corner case 1)
➔ date -r non-existent-file (Corner case 2)
➔ date -r (Corner case 3)

rm

This command allows the user to delete files using the command-line.
Format: $ rm [-v/-i] <file>

Options

1. -v: verbose (lets you know what file you removed)
2. -i: Seek confirmation before proceeding with remove

Corner Cases

1. Handle the case of using rm on directories
2. Handle the case where an invalid file name is provided as argument

Assumptions

1. Removing multiple files at once is not supported

Testcases

➔ rm a.txt

- ➔ rm -v b.txt
- ➔ rm -i c.txt
- ➔ rm cmdbin (Corner case 1)
- ➔ rm non-existent-file (Corner case 2)
- ➔ rm

mkdir

This command allows the user to make new directories as long as the directory does not already exist.
Format: $ mkdir [-v/-p] <directory> <...directories>

Options

1. -v: verbose (lets you know what directory you created)
2. -p: Creates intermediate directories as required

Corner Cases

1. Create multiple directories at once (space-separated)
2. Handle the case where the user tries to create an already existing directory

Assumptions

1. Use of options is not supported when creating multiple directories at once

Testcases

- ➔ mkdir hello123
- ➔ mkdir -v hello1234
- ➔ mkdir -p hello1/hello2/hello3
- ➔ mkdir dir1 dir2 dir3 (Corner case 1)
- ➔ mkdir hello123 (Corner case 2)

Other Features

1. Thread-based Execution

   Append &t to any external command for thread-based execution

2. Colored shell prompt for aesthetics

References

https://man7.org/linux/man-pages
https://www.geeksforgeeks.org/multithreading-c-2/
https://stackoverflow.com/questions/4842424/list-of-ansi-color-escape-sequences/33206814#33206814
https://pubs.opengroup.org/onlinepubs/009695399/basedefs/sys/stat.h.html
https://pubs.opengroup.org/onlinepubs/009695399/basedefs/sys/types.h.html#tag_13_67