

# CS4218 Project - Milestone 2

## Assumptions

### Commands

#### General

- All command line arguments contain at least 1 non-whitespace character.
- All command line arguments do not contain leading or trailing whitespace.
- All command line arguments that are longer than 1 character and starts with a `-` are treated as options.
- All file and directories do not start with `.`.
- There are no symbolic links in the environment.
- Files and streams are correctly terminated and are not malformed.
- No emojis or special characters in file/directory names and contents.

#### CallCommand

- Command tokenization and parsing of special characters (redirection, pipe, quotes etc.) closely follow the behavior of the Unix shell GNU `bash`, version `3.2.57(1)-release (x86_64-apple-darwin17)` unless specifically stated in the Project Description.
- For example, the command `< a < b cat` has a valid syntax, and will work correctly if the files `a` and `b` exist.

#### Command Substitution

- **Each newline** in the subcommand output is replaced with **one space character**, as specified in the project description. This might make the output slightly surprising in the case of back quotes nested in double quotes.
  - For example, `echo "`echo hello ; echo ; echo world`"` will evaluate to `hello world` (note the two spaces in the middle and the one space at the end). Each of the three spaces in the output corresponds to a newline that trails each `echo` command.

### Applications

#### CdApplication

- The root directory is `/`.
- Absolute paths refer to paths that start with a `/`.

- Paths can one or more of the contain the following:
  - `..` - refers to one directory up
  - `.` - refers to the current directory
- `cd` throws exception when the number of arguments is not one.
- `cd` does not use the `InputStream` and `OutputStream`
- Not supporting `cd -` and `cd ~` (and the equivalent `cd`)

## CmpApplication

- Extended Ascii table characters are acceptable

## GrepApplication

- `grep` without piping an inputstream, `grep <pattern>` is not supported.

## ExitApplication

- To exit the shell, the exit application sends a signal to the shell by throwing an `ExitException`. This exception will be caught by `Shell` and `Shell` will end its execution.

## LsApplication

- Input arguments are not escaped, i.e. no spaces
- Hidden files are ignored
- The only valid argument of the application is directory. (Using `ls` with files or symbolic links will result in an error message).
- If the user supplied at least one directory as an argument to `ls`, the result of the application will be as follows:

```
<directory path relative to current directory>:
<contents of directory>
```

## MkdirApplication

- Intermediate directories that do not exist will not be created and will throw exception if attempted
- Application is not failfast. Will create as many legal inputs as possible while noting down those that failed and throw exception after operation completes
- Max filename length of 255

## PasteApplication

- As in bash, directories are treated as empty file.
  - `paste <directory>` prints an empty string to the output stream.
- `paste -` without piping inputstream is not supported.

## SplitApplication

- No limit on the number of splits
- Split result ends up in current directory and not the directory of the origin file
- Overwrites files with same name

## SedApplication

- Replacement index has to be a positive, non-zero number
- There is no 'global' option that replaces **all occurrences** in a single line