

Unix Shell

Project Description:

- Create a Unix Shell that takes an input command, parses it out, and executes the command.

Standards Operating Procedure:

1. User enters command
2. Main gets the command (`cin.getline(...)`) then saves a copy of the command to `last_command`
3. Main then calls `parse_command` from the `UNIXShell` class to 'break' the command into individual words and returns the number of words counted (`int count;`)
 - a. `parse_command` checks if "&" is the last character in the command line.
 - i. If "&" is present `should_wait` is set to *false*
 - See `parse_command()`
4. Main checks if the command is a "reserved command"
 - a. If a reserved command `exec_reserved_command()` is called
 - Eg. "exit" or "!!"
 - See `exec_reserved_command()`
5. `exec_command()` is called from `UNIXShell` to execute the command.
6. Process repeats until "exit" is entered

UNIXShell properties:

- `should_wait`
 - Used to determine if child and parent process should run concurrently
 - Boolean; *true* by default
 - *private*
- `last_command`
 - Copy of previous command
 - String; Empty by default
 - *private*
- `Should_run`
 - Used to determine if shell program should continue running
 - Boolean; *true* by default
 - *public*

UNIXShell methods:

- `parse_command()`
 - Description:
 - Separates command line entry into individual words then adds them to `args[]`.
 - Sets `should_wait` to *false* when "&" is at end of command line
 - Used at:
 - `main()`

- `exec_reserved_command()`.
- `exec_command()`
 - Description:
 - Used to execute the command line.
 - Creates `fork()` of itself. If the process ID (`pid_t pid;`) is 0 (zero; is a child process) it analyzes `args[]`.
 - If count is 3 or more it checks if special characters are used.
 - Eg. "<" or ">"
 - See `redirect_file_out()` or `redirect_file_in()` for more information
 - `execvp()` is called to execute the actual command
 - If the command is not recognized returns -1.
 - Is a Linux command
 - `Should_wait`
 - **true:** Parent process waits for child processes to finish before running
 - **false:** The parent process runs while child processes run concurrently
 - Used at:
 - `main()`
 - `exec_reserved_command()`
- `exec_reserved_command()`
 - Description:
 - If user enters special command that command is executed
 - Special commands:
 - "exit" to end the program
 - Shell process ends
 - `should_run = false`
 - "!!" executes previous command
- `save_last_command()`
 - Description:
 - To copy `command[]` so other methods of `UNIXShell` can access it
 - Used at:
 - `main()`
- `is_reserved_command()`
 - Description: Checks if the command is "exit" or "!!"
 - "exit" or "!!" returns false
- `redirect_file_out()`
 - Redirects terminal output to specified file name
 - When ">" is used `exec_command()` calls this method
- `redirect_file_in()`
 - Redirects terminal input from specified file name
 - When "<" is used `exec_command()` calls this method