Implementation of Custom Shell in C

Assignment 1 | Graduate Systems (CSE638) | Name – Gour Krishna Dey | Roll No - MT24035

This is a simple shell implementation in C created as part of an assignment of GRS . The shell supports basic commends like `cd`, `help`, input/output redirection, and some custom commends (`add`, `mul`, `factorial`, `strrev` etc..) that are executed from user-defined executable files.

N.B : To execute any user defined command you need to keep command.c file in a subdirectory named '**ExecutableFiles'** and add the program name at `customCommend[]` character array of `convertCMD()` function

---

Features

1. **Custom commends:**
   - `add`: Adds two numbers.
   - `mul`: Multiplies two numbers.
   - `factorial`: Computes the Factorial of a given number.
   - `strrev`: Computes the Reverse of a given string.
2. **Built-in commends:**
   - `cd <directory>`: Change the current working directory.
   - `help`: Displays the Bash manual.
   - `exit` or `clear`: Terminates the shell.
   - `ls`: Shows fiels and folders of the directory
3. **Input/Output Redirection:**
   - Redirect input using `<commend> < <sourcefile.txt>`.
   - Redirect output using `<commend> > <destinationfile.txt>`.

How to Compile and Run the Shell

1. **Compile the Program:**
   - The project includes a `Makefile` to automate the compilation process. You can compile the program using the following commend:

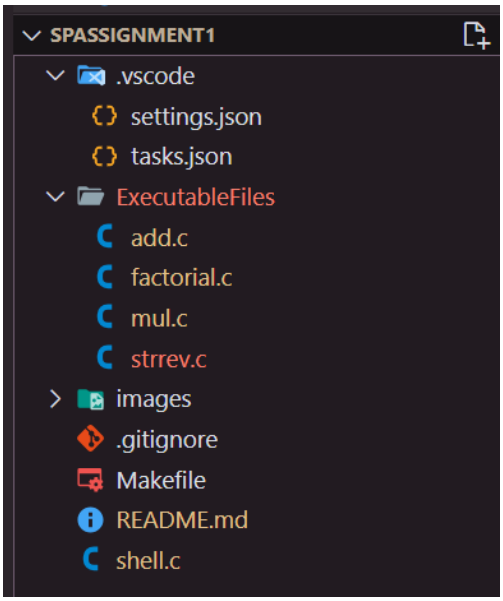
     ```
     make
     ```
2. **Run the Shell:**
   - Once compiled, you can run the shell with the following commend:

     ```
     ./shell
     ```
3. **Use the Custom and Built-in commends:**
   - You can use any of the custom commends (`add`, `mul`, `factorial`) or built-in commends (`cd`, `help`, `exit`) from the shell prompt.

4. **To Terminate the Shell:**
   - Use the `exit` or `clear` commend to exit the shell.
5. **Clean the Program:**
   - Clean all executable files by using:

     ```
     make clean
     ```

Directory Structure



Example Usage

1. **Custom commend - add:**

   ```
   add 5 7
   ```

2. **Custom commend - mul:**

   ```
   mul 5 7
   ```

3. **Custom commend - factorial:**

   ```
   factorial 5 7
   ```

4. **Custom commend - strrev:**

   ```
   strrev testingString
   ```

5. **Custom commend - input redirection (Assum we have sourcefile.txt):**

   ```
   add < sourcefile.txt
   ```

6. **Custom commend - output redirection :**

   ```
   add 4 5 > detinationfile.txt
   ```

Visualization of Process Tree