**Assignment 0 Part 2**

**Create a small Shell using C language**

**NAME : HARDIK SAINI**

**ROLL NO. : 2018391**

**C learning Sources.**

Link to YouTube playlist for learning pointer in C :- <https://www.youtube.com/watch?v=h-HBipu_1P0&list=PL2_aWCzGMAwLZp6LMUKI3cc7pgGsasm2_>

Link to YouTube playlist to learn basic Syntax :- <https://www.youtube.com/watch?v=CCZE4zuOwZM&list=PLS1QulWo1RIZlA5oGczk8kY7Eenytc33s>

**Sources for building Shell.**

<https://stackoverflow.com/>



<https://www.geeksforgeeks.org/>

<https://www.youtube.com/watch?v=PwxTbksJ2fo>

<https://www.youtube.com/watch?v=9seb8hddeK4>

<https://www.youtube.com/watch?v=fz2jhcTltdw>

**Pre-requisite in order to perform error free compilation (Make sure that there is enough memory present to fork a child process)**

1. **See the headers included in file. Make sure that you have installed all the required modules/packages before including them else it would give error that module not found below are the all headers that included in file**

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<unistd.h>

#include<sys/types.h>

#include<sys/wait.h>

make sure that make is installed **sudo apt-get install make**

1. **If the packages are not installed then the program won’t compile**
2. **Upon extracting the zipped file open the folder 2018391\_CAOS\_Shell it would be containing**
3. **basicFunctions.c**
4. **basicFunctions.h**
5. **OuterCommandHandler.c**
6. **OuterCommandHandler.h**
7. **shell.c**
8. **Makefile**
9. **Open terminal here and execute command “make”**
10. **Upon executing command if you have installed all packages correctly it would have created a file “Shell” without extension file**
11. **Execute Shell file by typing ./Shell**
12. **now Shell is running…**

**Assumption is that you’re executing make file in same directory.**

**COMMANDS DESCRIPTION**

|  |  |  |
| --- | --- | --- |
| **COMMAND NAME** | **FLAGS** | **DESCRIPTION** |
| **echo** | [**\\n**](file:///\\n) | **New line** |
| [**\\b**](file:///\\b) | **Backspace** |
| [**\\c**](file:///\\c) | **Suppress Output** |
| **history** | **-c** | **Clear history** |
| **-l** | **First 5 commands** |
| **pwd** | **Flags not available** | **Current working directory** |
| **exit** | **Flags not available** | **Exit from shell** |
| **cd** | **Flags not available** | **Changes directory** |
| **ls** | **-l** | **Shows list of all files present** |
| **-a** | **Show all files (hidden too)** |
| **cat** | **--help (No other flag available)** | **Opens help** |
| **--version (No other flag available)** | **Opens Version of current cat** |
| **date** | **--date={option} (multiple)** | **Displays date according to option** |
| **--help** | **Opens help** |
| **--version** | **Opens Version of date** |
| **rm** | **-i** | **Asks before deleting** |
| **-r** | **Deletes recursively** |
| **mkdir** | **--help (Flags not available)** | **Display help** |
|  | **--version (Flags not available)** | **Display version** |
| **clear** | **No flags available** | **Clear output screen** |

**Test cases**

1. **echo apple**
2. **echo apple\\nbanana**
3. **echo appe\\ble**
4. **history**
5. **history -l**
6. **history -c**
7. **pwd**
8. **cd /bin**
9. **exit**
10. **ls**
11. **ls -l**
12. **ls -a**
13. **ls /**
14. **clear**
15. **cd /**
16. **cat /etc/apt/sources.list**
17. **cat --help**
18. **cat --version**
19. **date --date=yesterday**
20. **date --date=tomorrow**
21. **date --date=1 month ago**
22. **date --date=1 year ago**
23. **date --date=1 year**
24. **date --help**
25. **date --version**
26. **mkdir 1**
27. **mkdir 1/2**
28. **mkdir 1/2/3**
29. **rm 1 (Error)**
30. **rm -r 1 (Will delete directories recursively)**
31. **rm {file name}**
32. **rm -i {filename}**

**Protection & defenses**

**I HAVE CREATED SHELL IN SUCH A WAY THAT IF USER EXECUTES SOMES UNKNOWN COMMAND THE SHELL WON’T CRASH RATHER WILL PRINT THE MESSAGE UNKNOWN COMMAND**

**PROTECTION AGAINST WRONG COMMAND**

**Example : ls cat**

**Error directory not found**