

Simulating Custom Linux Commands

Tanaya Gogawale (GR no. 11910384)

Abstract:- Project is about creating custom new linux commands which are not already present in the system. Commands runs on the inbuilt linux terminal and present in the system so can be used anytime on the particular system having commands added in system. This reduces the efforts of programmer to write explicit code and programmer can later delete or have modifications in command.

Keywords:- Linux, Terminal, User Defined, Arithmetic, Logical, Fun, File Handling, Searching, Sorting, Trigonometric, Puzzle, Commands, CPP.

I. INTRODUCTION

“Custom Linux Commands” is a project which includes a set of commands that are written in CPP. These commands are only for Linux operating system. As Linux allows the user to interfere in the system, we can make these commands and access them from the inbuilt system terminal anytime without the need of installing the package. The project includes commands that are useful for any programmer in his daily life. Commands have easy syntax to remember and operate.

II. IMPLEMENTATION DETAILS

Custom Linux Commands are the commands for the linux operating system’s terminal. These commands are made in C++ programming language.

Tools and Technologies:

CPP:

Procedural, high-level, Object-Oriented, Programming Language, simple.

Linux operating system:

Portable, Open source, secure, Multi-user.

Notepad:

Simple text editor, allows programming from scratch.

GCC Compiler:

pre-processing, compilation, assembly and linking of source code to generates an executable file.

List of Commands:

- Basic:
n_hello
n_help

- Arithmetic:

n_sub
n_add
n_mul
n_pow

- Trigonometric:

sin cos tan
cot sec cosec

- Editing:

n_uppercase
n_lowercase
n_sentencecase

- Sorting:

n_sortChar
n_sortNum

- Searching:

n_searchnum

- Fun:

n_randomjoke
n_puzzles

- File handling:

n_createfile
n_enterdata
n_deletefile

III. RESULTS AND DISCUSSION

Commands are divided into 10 major types. Given below are commands implemented. Till now total 23 commands are added in the system.

```

root@tanaya:~# n_help
Here is the list of commands
1.n_add(a,b)
2.n_sub(a,b)
3.n_mul(a,b)
4.n_div(a,b)
5.n_mod(a,b)
6.n_pow(a,b)
7.Rhythmbox -b,c..
8.Rhythmbox a ln(x,y,z..)
9.n_randomjoke
10.n_puzzle
11.n_indata
12.n_deldata
root@tanaya:~#

```

1. basic command

```
tanaya@tanaya:~$ n_add
Enter numbers:
47 290
Result: 337
tanaya@tanaya:~$ n_sum
21 -891
-870
tanaya@tanaya:~$ n_mul
Enter numbers:
28 29
Result: 812
tanaya@tanaya:~$ n_pow
34 2.3
3329.67
tanaya@tanaya:~$
```

2. Arithmetic Commands

```
tanaya@tanaya:~$ n_sortChar
ha wqh akjqw
>>akjqw ha wqh
tanaya@tanaya:~$ n_sortNum
E Files the size of array:3
Array:1 83 21 290 2
>>1 21 83
tanaya@tanaya:~$
```

5. Sorting commands

```
tanaya@tanaya:~$ sin
Angle in degrees: 73
>>0.9563
tanaya@tanaya:~$ cos
Angle in degrees: 37
>>0.799
tanaya@tanaya:~$ tan
Angle in degrees: 88
>>28.6
tanaya@tanaya:~$ cot
Angle in degrees: 82
>>0.141
tanaya@tanaya:~$ sec
Angle in degrees: 38
>>1.27
tanaya@tanaya:~$ cosec
Angle in degrees: 32
>>1.89
tanaya@tanaya:~$
```

3. Trigonometric commands

```
tanaya@tanaya:~$ n_searchnum
Enter the size of array:4
Array:82 231 23 2
Enter element to search:3
E Files t is not present
tanaya@tanaya:~$
```

6. Searching commands

```
tanaya@tanaya:~$ n_uppercase
hello
HELLO
tanaya@tanaya:~$ n_lowercase
Hello
hello
tanaya@tanaya:~$ n_sentencecase
: hey There..guD morning
>> Hey There..GuD morning
tanaya@tanaya:~$
```

4. Editing commands

```
tanaya@tanaya:~$ n_randomjoke
The worst thing about a Boolean is that even if you are wrong, you are only off by a bit.
tanaya@tanaya:~$ n_puzzles
There was a green house. Inside the green house there was a white house. Inside the white house there was a red house. Inside the red house there were lots of babies. What is it?
Home
Oops!!! Wrong Choice ...Better luck next time
tanaya@tanaya:~$
```

7.
8. Fun commands

```

tanaya@tanaya: ~/Desktop/n_commands
tanaya@tanaya:~/Desktop/n_commands$ n_createfile
Enter the Name of File: tanaya.txt
File Doesn't Exist!
C Text Editor File...
File Created Successfully!
tanaya@tanaya:~/Desktop/n_commands$ n_deletefile
Enter the Name of File: tanaya.txt
File deleted successfully
tanaya@tanaya:~/Desktop/n_commands$ n_enterdata
Enter the Name of File: file.txt
File Doesn't Exist!
Creating the File...
File Created Successfully!
Enter the Data: hello
have a nice day

```

9. File Handling

IV. CONCLUSION AND FUTURE SCOPE

Conclusion:

Creating new commands which are important in daily life made my concepts likes searching, sorting, data handling and programming skills stronger. Also, it helped me to get familiar with most of the linux commands and operating system.

Having explicit commands other than application within system makes the things easier because programmer can access it anytime without need of installing any API.

Future Scope:

Entering the large amount of data in the terminal may be messy and can cause errors, so creating write way to deal with such data.

Security is also important while dealing with files so to provide security before accessing selected commands. Add more user-friendly commands.

REFERENCES

- [1]. https://www.researchgate.net/publication/313315743_Introduction_to_Linux_and_Command_Line_Tools_for_Bioinformatics_Principles_and_Methods
- [2]. <https://linuxtogether.org/creating-custom-commands-in-unix-linux>
- [3]. https://www.researchgate.net/publication/228790967_Some_useful_linux_commands
- [4]. <http://www.cplusplus.com/>
- [5]. <https://youtu.be/sIw800iEFho>
- [6]. <https://youtu.be/5sPXQ0NtFd4>