Operating System
<Lab #1_Simple Shell >

Prepared Date :

Oct 15, 2019

Prepared by :

Hagar Usama 4970

Index :

Simple Shell

Requirements Specification

To run the program, you need a c++ compiler (Recommended : g++)

Design

The design is pretty simple; a simple shell 'Creamy shell' to get your commands and calls execvp to execute them.

Code Description

The code is in cpp language "shell.cpp" . One class is implemented to hold almost everything.

Part I (Creamy_Shell class):

```
class Creamy_Shell{

      public:
      Creamy_Shell();
    //runs the shell
      void run_shell();
      void greet();

      private:
      string command;
      vector<string> args;
      //splits parameters of command string
      void split_parameters();
      //trims spaces from command string
```

```
    void trim_command();
    //converts the vector of args into array of char**
    void get_arg_list(char* arr[]);
    //calling fork to create a new process
    void call_fork(bool state);
    //call cd command since no need for forking here
    void call_cd();

    };
```

run_shell() : here command is taken , parameters are split , and commands are checked to assign the proper execution.

greet() : a nice function to greet my users

Part II (Other functions) :

```
void trim(string &str);
// left trim
void ltrim(string &exp);
// right trim
void rtrim(string &exp);
// extract a certain expression & replaces it by a delimiter
string extract(string &exp , string re , string delim="");
//get matched expression from a string using regex
int get_matched(string s , regex reg , string &mat);

//write in log file (append)
void write_log(string filename , string str);
//write in a file (w)
void write_dic(string filename , string str);


//signal handler for terminated processes
void sweety_handler(int signal){
```

```
      pid_t pid;
     pid = wait(NULL);

      if(pid == -1)write_log("log.txt", "Child process was terminated
[foreground]\n");
      else write_log("log.txt", "Child process was terminated
[Background]\n");

      }
```

Trim functions : are just for trimming the input string

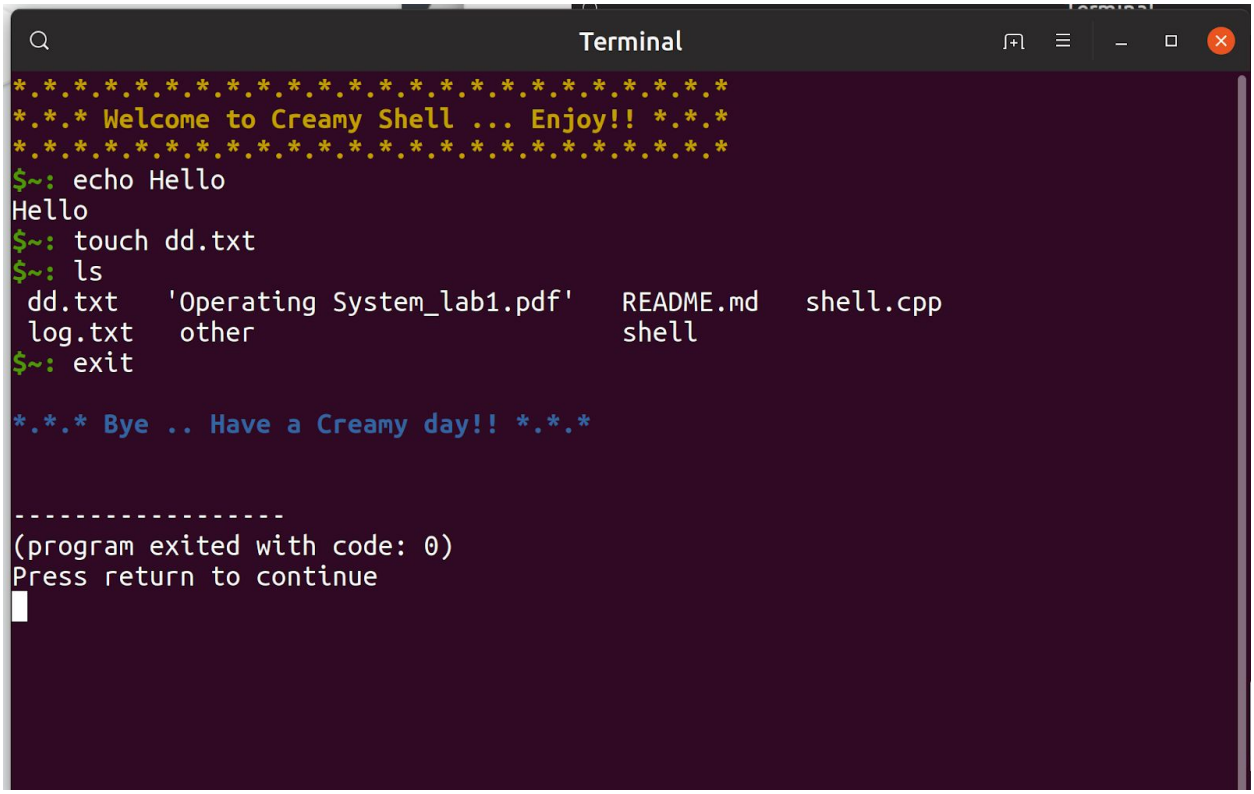
extract : returns a string from the given string based on the regular expression provided, and replaces the substring with a delimiter.

get_matched: searches for matches in the given string using regular expressions and returns 1 if found.

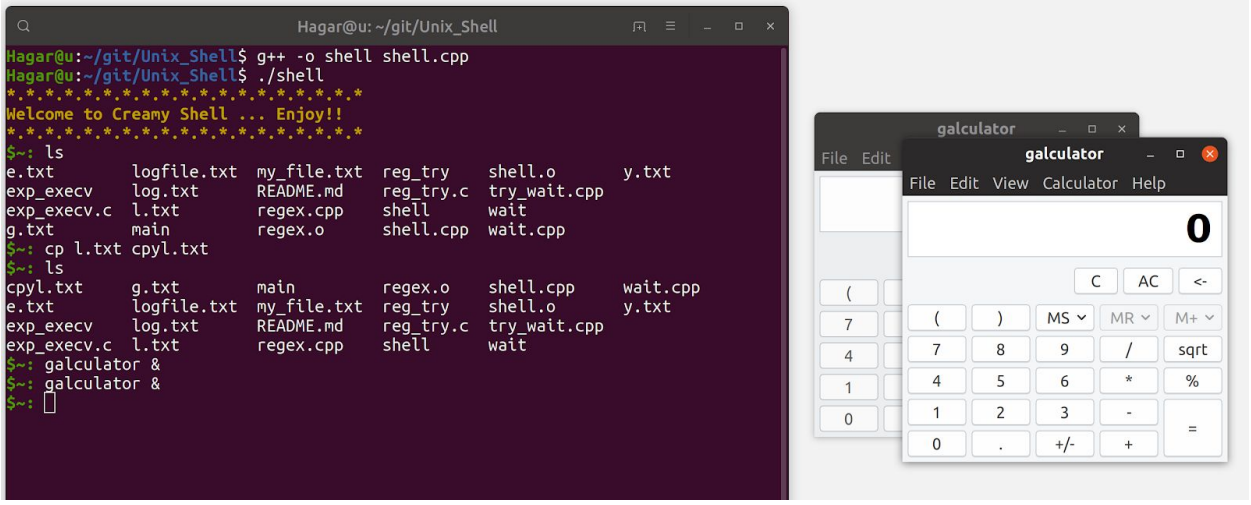
Write_functions: for history log of processes.

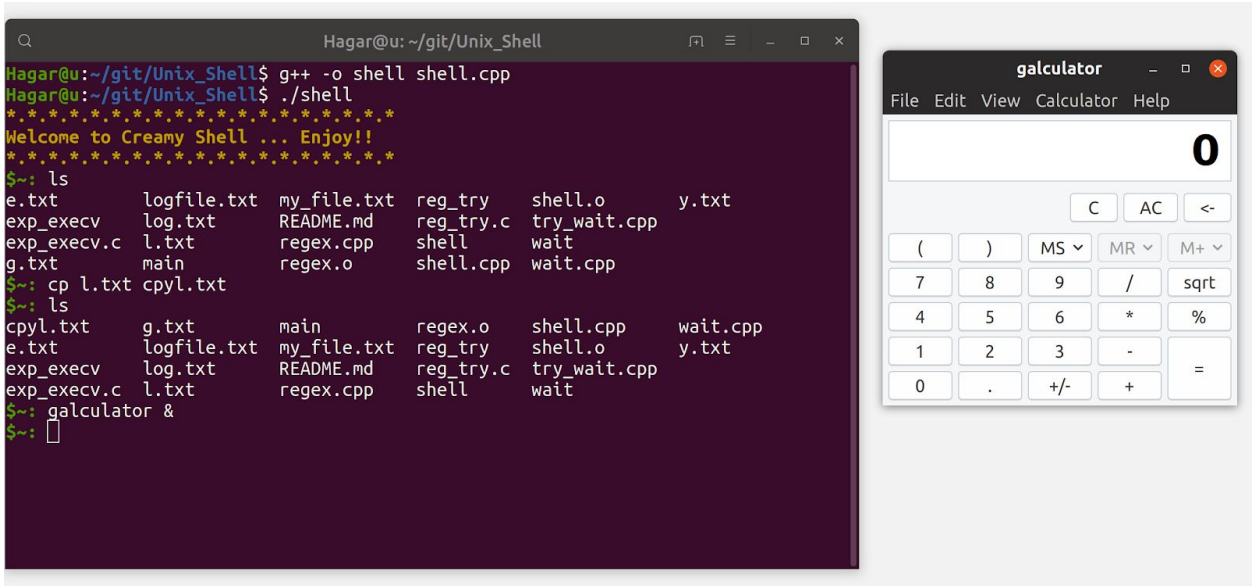
sweety_handler: the handler at which some actions will be executed (write_log) when getting response form signal function (after a child process is terminated)

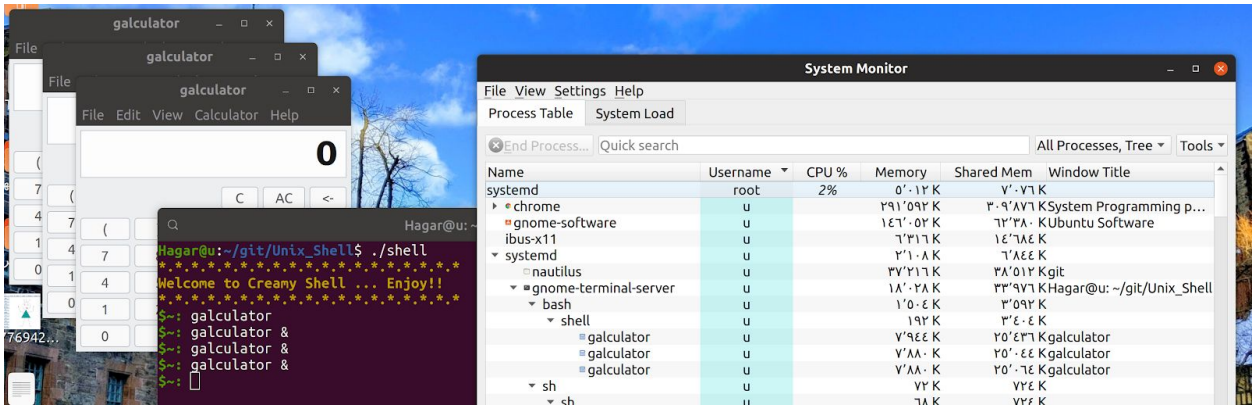
Sample Runs


Terminal
.
..* Welcome to Creamy Shell ... Enjoy!! *.*.*
.
$~: echo Hello
Hello
$~: touch dd.txt
$~: ls
dd.txt 'Operating System_lab1.pdf' README.md shell.cpp
log.txt other shell
$~: exit
..* Bye .. Have a Creamy day!! *.*.*

(program exited with code: 0)
Press return to continue


Hagar@u: ~/git/Unix_Shell
Hagar@u:~/git/Unix_Shell$ g++ -o shell shell.cpp
Hagar@u:~/git/Unix_Shell$ ./shell
.
Welcome to Creamy Shell ... Enjoy!!
.
$~: ls
e.txt logfile.txt my_file.txt reg_try shell.o y.txt
exp_execv log.txt README.md reg_try.c try_wait.cpp
exp_execv.c l.txt regex.cpp shell wait
g.txt main regex.o shell.cpp wait.cpp
$~: cp l.txt cpyl.txt
$~: ls
cpyl.txt g.txt main regex.o shell.cpp wait.cpp
e.txt logfile.txt my_file.txt reg_try shell.o y.txt
exp_execv log.txt README.md reg_try.c try_wait.cpp
exp_execv.c l.txt regex.cpp shell wait
$~: galculator &
$~: galculator &
$~:
galculator
File Edit
galculator
File Edit View Calculator Help
0
C AC <-
( ) MS MR M+
7 8 9 / sqrt
4 5 6 * %
1 2 3 - =
0 . +/- +


Hagar@u: ~/git/Unix_Shell
Hagar@u:~/git/Unix_Shell$ g++ -o shell shell.cpp
Hagar@u:~/git/Unix_Shell$ ./shell
.
Welcome to Creamy Shell ... Enjoy!!
.
$~: ls
e.txt logfile.txt my_file.txt reg_try shell.o y.txt
exp_execv log.txt README.md reg_try.c try_wait.cpp
exp_execv.c l.txt regex.cpp shell wait
g.txt main regex.o shell.cpp wait.cpp
$~: cp l.txt cpyl.txt
$~: ls
cpyl.txt g.txt main regex.o shell.cpp wait.cpp
e.txt logfile.txt my_file.txt reg_try shell.o y.txt
exp_execv log.txt README.md reg_try.c try_wait.cpp
exp_execv.c l.txt regex.cpp shell wait
$~: galculator &
$~:
galculator
File Edit View Calculator Help
0
C AC <-
( ) MS MR M+
7 8 9 / sqrt
4 5 6 * %
1 2 3 - =
0 . +/- +


galculator
File Edit View Calculator Help
0
Hagar@u:~/git/Unix_Shell$ ./shell
Welcome to Creamy Shell ... Enjoy!!
$~: galculator
$~: galculator &
$~: galculator &
$~: galculator &
System Monitor
File View Settings Help
Process Table
System Load
End Process...
Quick search
All Processes, Tree
Tools
Name Username CPU % Memory Shared Mem Window Title
systemd root 2% ٥٬٠١٢ K ٧٬٠٧١ K
chrome u ٢٩١٬٥٩٢ K ٣٠٩٬٨٧١ K System Programming p...
gnome-software u ١٤٦٬٠٥٢ K ٦٢٬٣٨٠ K Ubuntu Software
ibus-x11 u ٦٬٣١٦ K ١٤٬٦٨٤ K
systemd u ٢٬١٠٨ K ٦٬٨٤٤ K
nautilus u ٣٧٬٢١٦ K ٣٨٬٥١٢ K git
gnome-terminal-server u ١٨٬٠٢٨ K ٣٣٬٩٧١ K Hagar@u: ~/git/Unix_Shell
bash u ١٬٥٠٤ K ٣٬٥٩٢ K
shell u ١٩٢ K ٣٬٤٠٤ K
galculator u ٧٬٩٤٤ K ٢٥٬٤٣١ K galculator
galculator u ٧٬٨٨٠ K ٢٥٬٠٤٤ K galculator
galculator u ٧٬٨٨٠ K ٢٥٬٠٦٤ K galculator
sh u ٧٢ K ٧٢٤ K