

Os2 Project

Pripoae - Șerbănescu Mihai

Șerban Ramona-Andreea

Ștefănescu Mihai Daniel

Florescu Robert

Ciobanu Bianca Nicola

Project Contents:

1. Project Task

2. Project code displayed in Notepad++

3. Starting the project in Ubuntu

4. The “Help” section

4.1 The “exit” command

4.2 The “calc” command

4.3 The “history” command

4.4 The “timer” command

4.5 The “web” command

4.6 The “history_rem” command

4.7 The “batch” command

5. Linux commands

6. Combined multiple commands

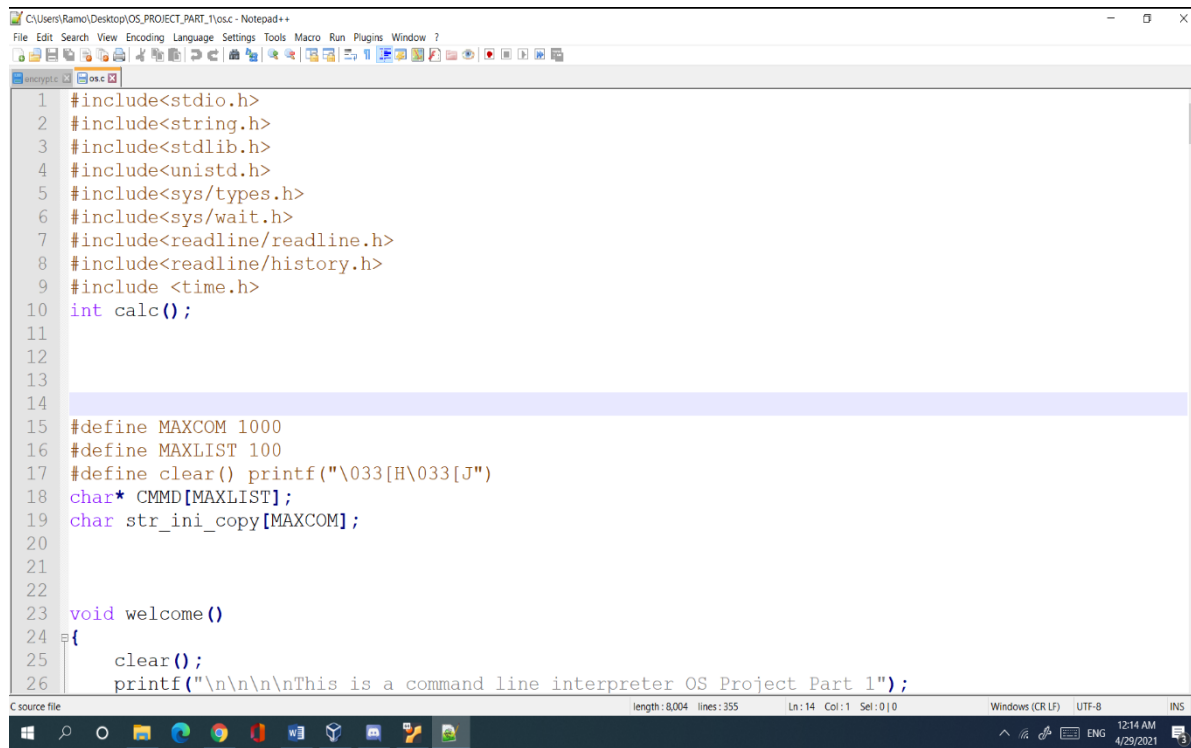
1. Project Task

Project Part 1

You are required by the OS company to write the code in C for implementing a command line interpreter. When typing a command, a child process is created and it executes the entered command. In some cases it asks the user to input the required data. The command line needs to manage multiple commands simultaneously.

The command line can be used interactively (waiting for input) or in batch mode (using a batch file which contains several commands). When the 'exit' keyword is introduced, all the processes which run are killed. Multiple commands can be run by separating them by the ';' character. Each of these commands will run simultaneously. The command line shouldn't output anything until all the commands which are executing have finished.

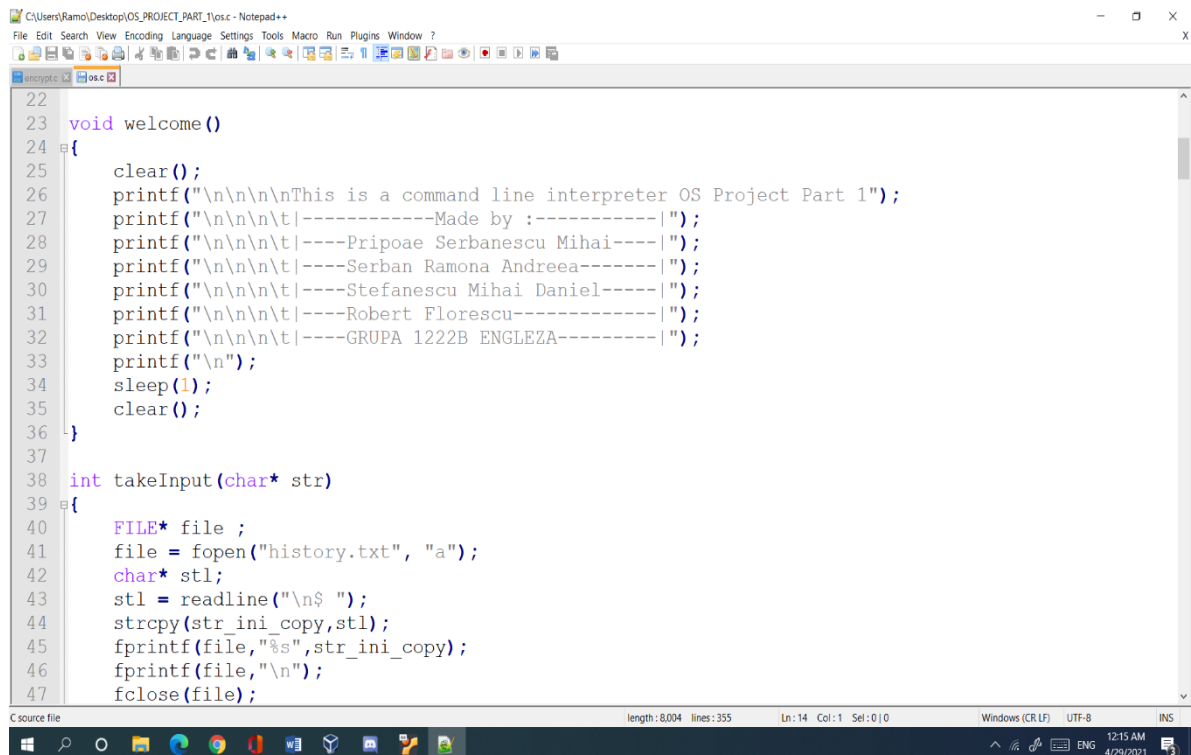
2. Project code displayed in Notepad++



The screenshot shows a Notepad++ window with the following C code:

```
1  #include<stdio.h>
2  #include<string.h>
3  #include<stdlib.h>
4  #include<unistd.h>
5  #include<sys/types.h>
6  #include<sys/wait.h>
7  #include<readline/readline.h>
8  #include<readline/history.h>
9  #include <time.h>
10 int calc();
11
12
13
14
15 #define MAXCOM 1000
16 #define MAXLIST 100
17 #define clear() printf("\033[H\033[J")
18 char* CMMD[MAXLIST];
19 char str_ini_copy[MAXCOM];
20
21
22
23 void welcome()
24 {
25     clear();
26     printf("\n\n\nThis is a command line interpreter OS Project Part 1");
```

The status bar at the bottom indicates: C source file, length: 8,004 lines: 355, Ln: 14 Col: 1 Sel: 0 | 0, Windows (CR LF), UTF-8, INS, 12:44 AM, 4/29/2021.



The screenshot shows the continuation of the C code in Notepad++:

```
22
23 void welcome()
24 {
25     clear();
26     printf("\n\n\nThis is a command line interpreter OS Project Part 1");
27     printf("\n\n\n\t|-----Made by :-----|");
28     printf("\n\n\n\t|----Priopae Serbanescu Mihai----|");
29     printf("\n\n\n\t|----Serban Ramona Andreea-----|");
30     printf("\n\n\n\t|----Stefanescu Mihai Daniel-----|");
31     printf("\n\n\n\t|----Robert Florescu-----|");
32     printf("\n\n\n\t|----GRUPA 1222B ENGLEZA-----|");
33     printf("\n");
34     sleep(1);
35     clear();
36 }
37
38 int takeInput(char* str)
39 {
40     FILE* file ;
41     file = fopen("history.txt", "a");
42     char* stl;
43     stl = readline("\n$ ");
44     strcpy(str_ini_copy,stl);
45     fprintf(file,"%s",str_ini_copy);
46     fprintf(file,"\n");
47     fclose(file);
```

The status bar at the bottom indicates: C source file, length: 8,004 lines: 355, Ln: 14 Col: 1 Sel: 0 | 0, Windows (CR LF), UTF-8, INS, 12:15 AM, 4/29/2021.

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\os.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c os.c
47 fclose(file);
48 if (strlen(stl) != 0) {
49     add_history(stl);
50     strcpy(str, stl);
51     return 0;
52 }
53 return 1;
54 }
55
56 void BATCH(){
57     char batch[1000];
58     printf("Type the thr path and name of the batch you want to execute(ex:/home/user/Desktop/batch.
59     scanf("%s",batch);
60     system(batch);
61 }
62
63 void printCMD_History(){
64     FILE* file;
65     file=fopen("history.txt","r");
66     char c;
67     c=fgetc(file);
68     while(c!=EOF){
69         printf("%c",c);
70         c=fgetc(file);
71     }
72 }
73
C source file length: 8,004 lines: 355 Ln: 14 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
12:16 AM
4/29/2021
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\os.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c os.c
71 }
72 }
73 void waitl( int sec )
74 {
75     clock_t end_wait;
76     end_wait = clock() + sec * CLOCKS_PER_SEC;
77     while (clock() < end_wait){}
78 }
79
80 void apocalypse(){
81     FILE* file ;
82     file = fopen("history.txt", "w");
83     fclose(file);
84 }
85
86 int timer(){
87     int hours,minutes, seconds, s;
88     time_t t;
89     struct tm *tm;
90
91     tm = localtime(&t);
92     printf("Enter the number of seconds: ");
93     scanf("%d", &seconds);
94     s = seconds;
95     while(seconds > 0){
96         printf("%02d:%02d:%02d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
97         s--;
98         if(s == 0){
99             tm = localtime(&t);
100             printf("%02d:%02d:%02d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
101             s = seconds;
102         }
103     }
104 }
105
C source file length: 8,004 lines: 355 Ln: 14 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
12:18 AM
4/29/2021
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\os.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c os.c
95 while(seconds > 0){
96     tm->tm_hour = s/3600;
97     tm->tm_min = (s-(3600*tm->tm_hour))/60;
98     tm->tm_sec = (s-(3600*tm->tm_hour)-(tm->tm_min*60));
99     printf("%02d:%02d:%02d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
100     s--;
101     seconds--;
102     wait(1);
103 }
104 return 0;
105 }
106
107 void printDir()
108 {
109     char PWD[1024];
110     getcwd(PWD, sizeof(PWD));
111     printf("\n---(CMD @ %s)---", PWD);
112 }
113
114 void CMD(char** chr, char str_ini_copy[MAXCOM])
115 {
116     pid_t pid = fork();
117     if (pid == -1) { printf("\nFork failed."); return; }
118     else if (pid == 0) { if (execvp(chr[0], chr) < 0) { printf("\n%s: Command not found", str_ini_copy); } exit(0); } else { wait(NULL); return; }
119 }
120
121 void CMDPiped(char** chr, char** chrpipe)
122 {
123     int pipe_V[2];
124     pid_t p1, p2;
125
126     if (pipe(pipe_V) < 0) { printf("\nPipe failed to initialize"); return; }
127     p1 = fork();
128     if (p1 < 0) { printf("\nFork failed"); return; }
129
130     if (p1 == 0) {
131         close(pipe_V[0]);
132         dup2(pipe_V[1], STDOUT_FILENO);
133         close(pipe_V[1]);
134         if (execvp(chr[0], chr) < 0) {
135             printf("\nCommand 1 could not be executed");
136             exit(0);
137         }
138     } else {
139         p2 = fork();
140         if (p2 < 0) { printf("\nFork failed"); return; }
141         if (p2 == 0) {
142             close(pipe_V[1]);
143             dup2(pipe_V[0], STDIN_FILENO);
144             close(pipe_V[0]);
145             if (execvp(chrpipe[0], chrpipe) < 0) {
146                 printf("\nCommand 2 could not be executed");
147                 exit(0);
148             }
149         } else {
150             wait(NULL);
151             wait(NULL);
152         }
153     }
154 }
155
156 void openWeb() {
157     char URL[1000];
158     printf("Type the url of the page you want to open\nURL: ");
159     scanf("%s", URL);
160     char CPY[2000];
161     strcpy(CPY, "xdg-open https://");
162     strcat(CPY, URL);
163     system(CPY);
164 }
165
166 void openHelp()
167 {
168     puts("\n---HELP---");
169     printf("Usage: %s <command>\n", argv[0]);
170 }
171
172 int main()
173 {
174     int seconds = 60;
175     struct tm *tm;
176     time_t s;
177     s = time(NULL);
178     tm = localtime(&s);
179     printDir();
180     CMD(argv, argv);
181     CMDPiped(argv, argv);
182     openWeb();
183     openHelp();
184     return 0;
185 }
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\os.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c os.c
128 if (p1 < 0) { printf("\nFork failed"); return; }
129 if (p1 == 0) {
130     close(pipe_V[0]);
131     dup2(pipe_V[1], STDOUT_FILENO);
132     close(pipe_V[1]);
133     if (execvp(chr[0], chr) < 0) {
134         printf("\nCommand 1 could not be executed");
135         exit(0);
136     }
137 } else {
138     p2 = fork();
139     if (p2 < 0) { printf("\nFork failed"); return; }
140     if (p2 == 0) {
141         close(pipe_V[1]);
142         dup2(pipe_V[0], STDIN_FILENO);
143         close(pipe_V[0]);
144         if (execvp(chrpipe[0], chrpipe) < 0) {
145             printf("\nCommand 2 could not be executed");
146             exit(0);
147         }
148     } else {
149         wait(NULL);
150         wait(NULL);
151     }
152 }
153
154 void openWeb() {
155     char URL[1000];
156     printf("Type the url of the page you want to open\nURL: ");
157     scanf("%s", URL);
158     char CPY[2000];
159     strcpy(CPY, "xdg-open https://");
160     strcat(CPY, URL);
161     system(CPY);
162 }
163
164 void openHelp()
165 {
166     puts("\n---HELP---");
167     printf("Usage: %s <command>\n", argv[0]);
168 }
169
170 int main()
171 {
172     int seconds = 60;
173     struct tm *tm;
174     time_t s;
175     s = time(NULL);
176     tm = localtime(&s);
177     printDir();
178     CMD(argv, argv);
179     CMDPiped(argv, argv);
180     openWeb();
181     openHelp();
182     return 0;
183 }
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\os.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c os.c
159 void openHelp()
160 {
161     puts("\n---HELP---"
162         "\nList of compatible commands:"
163         "\n>-All general UNIX shell commands"
164         "\n>-exit: Closes the script"
165         "\n>-calc: Simple calculator arithmetic operations."
166         "\n>-history: prints on screen command history from previous sessions and present session."
167         "\n>-timer: starts a countdown timer."
168         "\n>-web: Opens a webpage of choice. Example: When prompted to enter the URL type www.google.com "
169         "\n>-history rem: erases history file without deleting it"
170         "\n>-batch: Executes all commands in a batch file. After beeing prompted to input the batch file type ex: /home/user/Desktop/batch.sh "
171         "\n>-It can also run multiple commands at the same time using (;) : ex: ls;pwd;timer "
172     );
173     return;
174 }
175
176
177 int ownCmdHandler(char** chr)
178 {
179     int NoOfOwnCmds = 10, i, switchOwnArg = 0;
180     char* ListOfOwnCmds[NoOfOwnCmds];
181     char* username;
182
183     ListOfOwnCmds[0] = "exit";
184     ListOfOwnCmds[1] = "cd";
185     ListOfOwnCmds[2] = "help";
186     ListOfOwnCmds[3] = "hello";
187     ListOfOwnCmds[4] = "calc";
188     ListOfOwnCmds[5] = "history";
189     ListOfOwnCmds[6] = "timer";
190     ListOfOwnCmds[7] = "web";
191     ListOfOwnCmds[8] = "batch";
192     ListOfOwnCmds[9] = "history_rem";
193 }
194
C source file length: 8,004 lines: 355 Ln: 14 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
12:21 AM 4/29/2021
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\os.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c os.c
194 for (i = 0; i < NoOfOwnCmds; i++) {
195     if (strcmp(chr[0], ListOfOwnCmds[i]) == 0) {
196         switchOwnArg = i + 1;
197         break;
198     }
199 }
200
201 switch (switchOwnArg) {
202 case 1:
203     printf("\nGoodbye\n");
204     exit(0);
205 case 2:
206     chdir(chr[1]);
207     return 1;
208 case 3:
209     openHelp();
210     return 1;
211 case 4:
212     username = getenv("USER");
213     printf("\nUser: %s", username);
214     printf("\n-Hello there.\n-General Kenobi...\n");
215     return 1;
216 case 5:
217     calc();
218     return 1;
219 case 6:
220     printCMD_History();
221     return 1;
222 case 7:
223     timer();
224     return 1;
225 case 8:
226     openWeb();
227     return 1;
228 case 9:
229     history_rem();
230     return 1;
231 }
232
C source file length: 8,004 lines: 355 Ln: 14 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
12:22 AM 4/29/2021
```

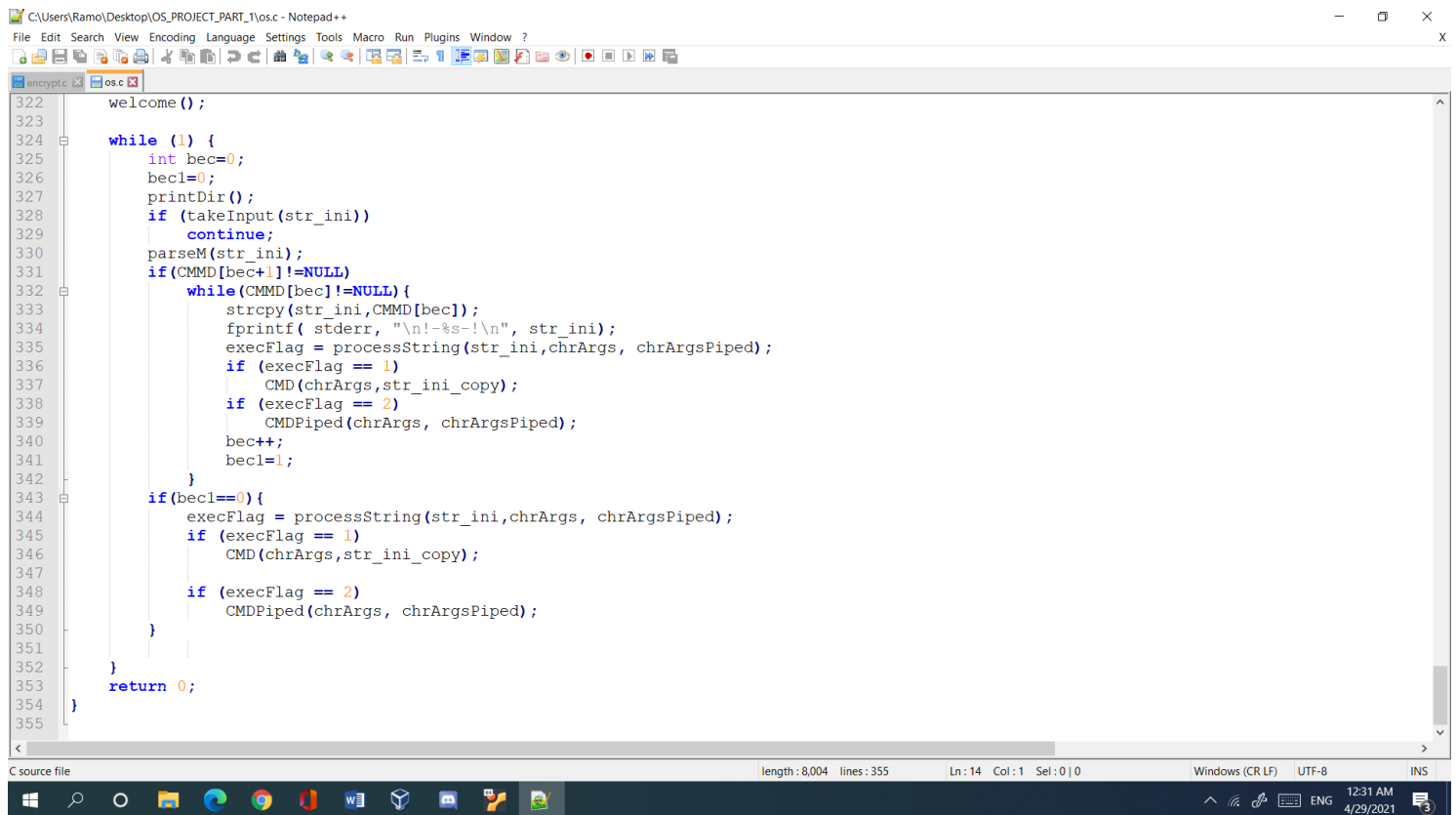
```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\osc - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encryptc osc
226     openWeb();
227     return 1;
228 case 9:
229     BATCH();
230     return 1;
231 case 10:
232     apocalypse();
233     return 1;
234
235     default:
236         break;
237 }
238
239 return 0;
240 }
241
242 int parsePipe(char* str, char** strpiped)
243 {
244     int i;
245     for (i = 0; i < 2; i++) {
246         strpiped[i] = strtok(&str, "|");
247         if (strpiped[i] == NULL) break;
248     }
249
250     if (strpiped[1] == NULL) return 0;
251     else return 1;
252 }
253
254 int parseM(char* z)
255 {
256     int i;
257     for (i = 0; i < 100; i++)
258         CMMD[i] = strtok(&z, ";");
259 }
260
C source file length: 8,004 lines: 355 Ln: 14 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
12:23 AM 4/29/2021
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\osc - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encryptc osc
261 void parseSpace(char* str, char** chr)
262 {
263     int i;
264     for (i = 0; i < MAXLIST; i++) {
265         chr[i] = strtok(&str, " ");
266         if (chr[i] == NULL) break;
267         if (strlen(chr[i]) == 0) i--;
268     }
269 }
270
271 int processString(char* str, char** chr, char** chrpipe)
272 {
273     char* strpiped[2];
274     int piped = 0;
275     piped = parsePipe(str, strpiped);
276
277     if (piped) {
278         parseSpace(strpiped[0], chr);
279         parseSpace(strpiped[1], chrpipe);
280     } else parseSpace(str, chr);
281
282     if (ownCmdHandler(chr)) return 0;
283     else return 1 + piped;
284 }
285
286 int calc()
287 {
288     int a, b, result;
289     char op;
290     printf("Enter first number: \n");
291     scanf("%d", &a);
292     printf("Enter arithmetic operator: \n");
293     scanf("%c", &op);
294     scanf("%d", &b);
295     result = a + b;
296     printf("Result: %d\n", result);
297 }
298
C source file length: 8,004 lines: 355 Ln: 14 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
12:28 AM 4/29/2021
```



```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\osc - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c osc
285
286 int calc()
287 {
288     int a, b, result;
289     char op;
290     printf("Enter first number: \n");
291     scanf("%d", &a);
292     printf("Enter arithmetic operator: \n");
293     scanf("%c", &op);
294     scanf("%c", &op);
295     printf("Enter second number: \n");
296     scanf("%d", &b);
297     switch(op)
298     {
299         case '+':
300             result = a + b;
301             break;
302         case '-':
303             result = a - b;
304             break;
305         case '*':
306             result = a * b;
307             break;
308         case '/':
309             result = a / b;
310             break;
311     }
312     printf("\nResult = %d", result);
313     return 0;
314 }
315
316 int main()
317 {
318     char str_ini[MAXCOM], *chrArgs[MAXLIST];
319     char* chrArgsPiped[MAXLIST];
320     int execFlag = 0;
321     int bec1=0;
322     welcome();
323
324     while (1) {
325         int bec=0;
326         bec1=0;
327         printDir();
328         if (takeInput(str_ini))
329             continue;
330         parseM(str_ini);
331         if(CMMD[bec+1]!=NULL)
332             while(CMMD[bec]!=NULL){
333                 strcpy(str_ini,CMMD[bec]);
334                 fprintf(stderr, "\n!-s-!\n", str_ini);
335                 execFlag = processString(str_ini,chrArgs, chrArgsPiped);
336                 if (execFlag == 1)
337                     CMD(chrArgs,str_ini_copy);
338                 if (execFlag == 2)
339                     CMDPiped(chrArgs, chrArgsPiped);
340                 bec++;
341                 bec1=1;
342             }
343         if(bec1==0){
344             execFlag = processString(str_ini,chrArgs, chrArgsPiped);
345             if (execFlag == 1)
346                 CMD(chrArgs,str_ini_copy);
347             if (execFlag == 2)
348                 CMDPiped(chrArgs, chrArgsPiped);
349         }
350     }
351 }
```

```
C:\Users\Ramo\Desktop\OS_PROJECT_PART_1\osc - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
encrypt.c osc
315
316 int main()
317 {
318     char str_ini[MAXCOM], *chrArgs[MAXLIST];
319     char* chrArgsPiped[MAXLIST];
320     int execFlag = 0;
321     int bec1=0;
322     welcome();
323
324     while (1) {
325         int bec=0;
326         bec1=0;
327         printDir();
328         if (takeInput(str_ini))
329             continue;
330         parseM(str_ini);
331         if(CMMD[bec+1]!=NULL)
332             while(CMMD[bec]!=NULL){
333                 strcpy(str_ini,CMMD[bec]);
334                 fprintf(stderr, "\n!-s-!\n", str_ini);
335                 execFlag = processString(str_ini,chrArgs, chrArgsPiped);
336                 if (execFlag == 1)
337                     CMD(chrArgs,str_ini_copy);
338                 if (execFlag == 2)
339                     CMDPiped(chrArgs, chrArgsPiped);
340                 bec++;
341                 bec1=1;
342             }
343         if(bec1==0){
344             execFlag = processString(str_ini,chrArgs, chrArgsPiped);
345             if (execFlag == 1)
346                 CMD(chrArgs,str_ini_copy);
347             if (execFlag == 2)
348                 CMDPiped(chrArgs, chrArgsPiped);
349         }
350     }
351 }
```



```
322 welcome();
323
324 while (1) {
325     int bec=0;
326     bec1=0;
327     printDir();
328     if (takeInput(str_ini))
329         continue;
330     parseM(str_ini);
331     if(CMMD[bec+1]!=NULL)
332         while(CMMD[bec]!=NULL){
333             strcpy(str_ini,CMMD[bec]);
334             fprintf(stderr, "\n!-s-!\n", str_ini);
335             execFlag = processString(str_ini,chrArgs, chrArgsPiped);
336             if (execFlag == 1)
337                 CMD(chrArgs,str_ini_copy);
338             if (execFlag == 2)
339                 CMDPiped(chrArgs, chrArgsPiped);
340             bec++;
341             bec1=1;
342         }
343     if(bec1==0){
344         execFlag = processString(str_ini,chrArgs, chrArgsPiped);
345         if (execFlag == 1)
346             CMD(chrArgs,str_ini_copy);
347
348         if (execFlag == 2)
349             CMDPiped(chrArgs, chrArgsPiped);
350     }
351 }
352
353 return 0;
354 }
355
```

3.Starting the project in Ubuntu

We need to copy the following command from the Makefile file:

- ***gcc -o exec os.c -lreadline***

In case it does not work, we must write the following line (gcc should also be installed):

- ***sudo apt-get install libreadline-dev***

We use ./exec to run it

Activities Terminal apr 29 01:31

ramona@ramona-VirtualBox: ~/Desktop

```
ramona@ramona-VirtualBox:~/Desktop$ gcc -o_exec os.c -lreadline
ramona@ramona-VirtualBox:~/Desktop$ ./exec
```

Activities Terminal apr 29 01:32

ramona@ramona-VirtualBox: ~/Desktop

```
This is a command line interpreter OS Project Part 1

|-----Made by :-----|

|----Pripoae Serbanescu Mihai----|

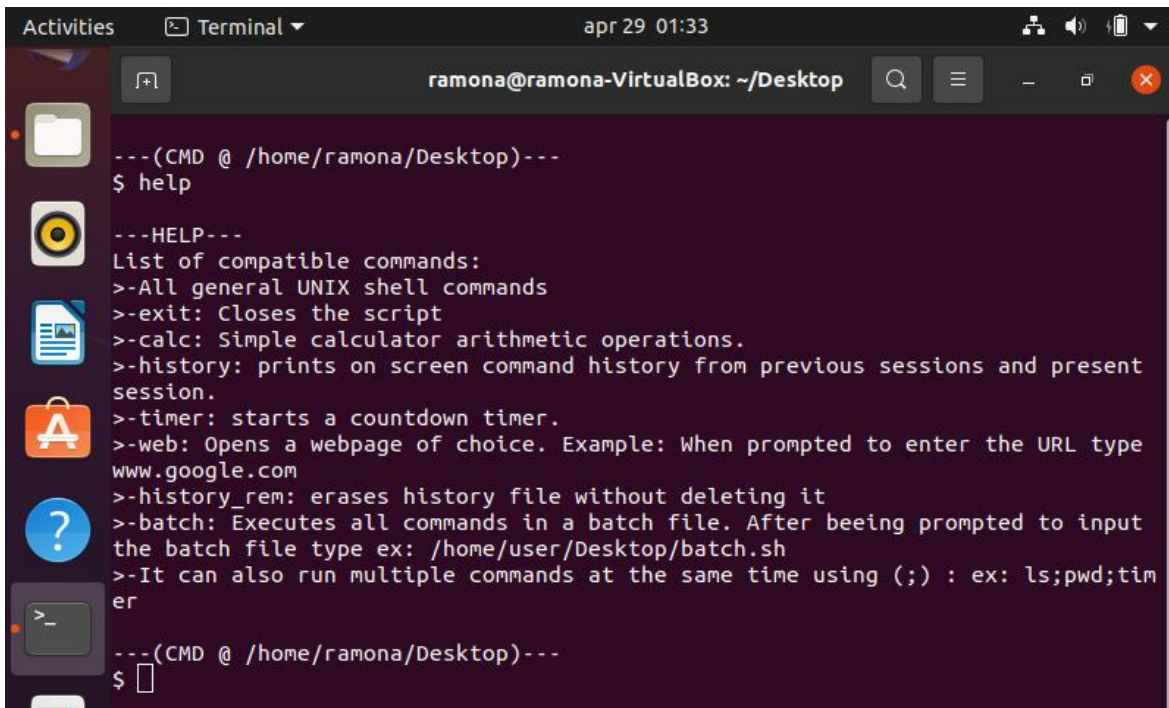
|----Serban Ramona Andreea-----|

|----Stefanescu Mihai Daniel-----|

|----Robert Florescu-----|

|----GRUPA 1222B ENGLEZA-----|
```

4. The “Help” Section



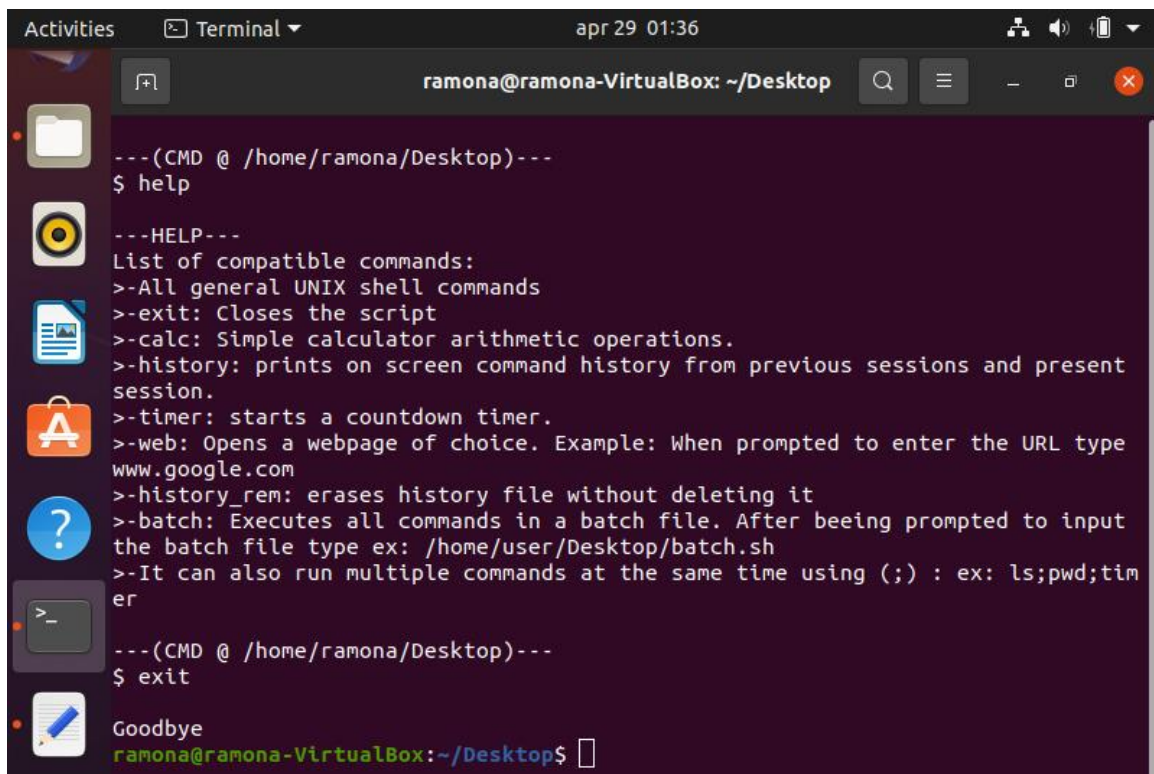
A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" showing the output of the "help" command. The window has a dark background with light-colored text. The left sidebar shows various application icons. The terminal output lists compatible commands and their functions.

```
---(CMD @ /home/ramona/Desktop)---
$ help

---HELP---
List of compatible commands:
>-All general UNIX shell commands
>-exit: Closes the script
>-calc: Simple calculator arithmetic operations.
>-history: prints on screen command history from previous sessions and present session.
>-timer: starts a countdown timer.
>-web: Opens a webpage of choice. Example: When prompted to enter the URL type www.google.com
>-history_rem: erases history file without deleting it
>-batch: Executes all commands in a batch file. After beeing prompted to input the batch file type ex: /home/user/Desktop/batch.sh
>-It can also run multiple commands at the same time using (;) : ex: ls;pwd;tim
er

---(CMD @ /home/ramona/Desktop)---
$
```

6.1 The “exit” command



A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" showing the output of the "exit" command. The window has a dark background with light-colored text. The left sidebar shows various application icons. The terminal output shows the "exit" command being entered and the resulting "Goodbye" message and prompt change.

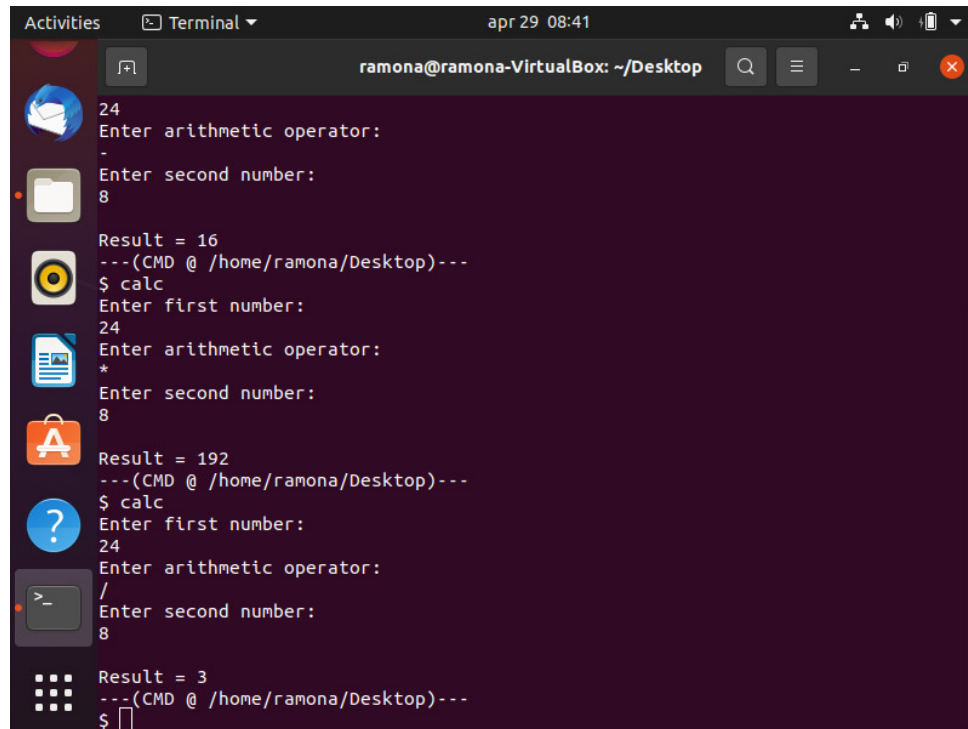
```
---(CMD @ /home/ramona/Desktop)---
$ help

---HELP---
List of compatible commands:
>-All general UNIX shell commands
>-exit: Closes the script
>-calc: Simple calculator arithmetic operations.
>-history: prints on screen command history from previous sessions and present session.
>-timer: starts a countdown timer.
>-web: Opens a webpage of choice. Example: When prompted to enter the URL type www.google.com
>-history_rem: erases history file without deleting it
>-batch: Executes all commands in a batch file. After beeing prompted to input the batch file type ex: /home/user/Desktop/batch.sh
>-It can also run multiple commands at the same time using (;) : ex: ls;pwd;tim
er

---(CMD @ /home/ramona/Desktop)---
$ exit

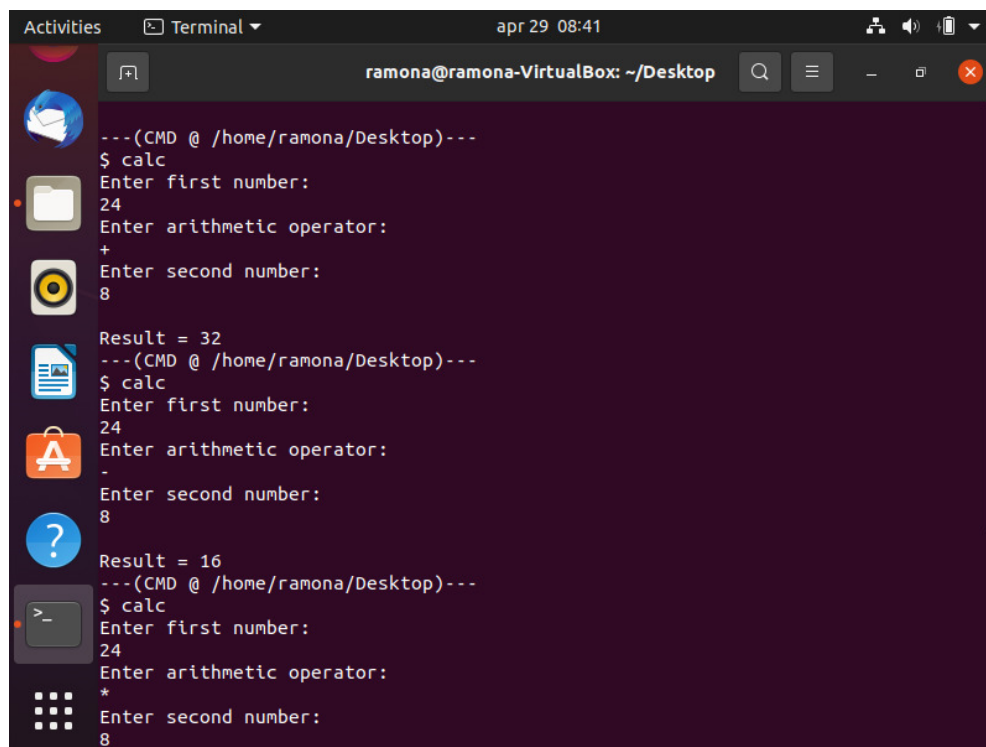
Goodbye
ramona@ramona-VirtualBox:~/Desktop$
```

6.2 The “calc” command



A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" showing the execution of the 'calc' command. The user enters '24' as the first number, '-' as the arithmetic operator, and '8' as the second number. The terminal displays "Result = 16". This process is repeated with '*' as the operator, resulting in "Result = 192".

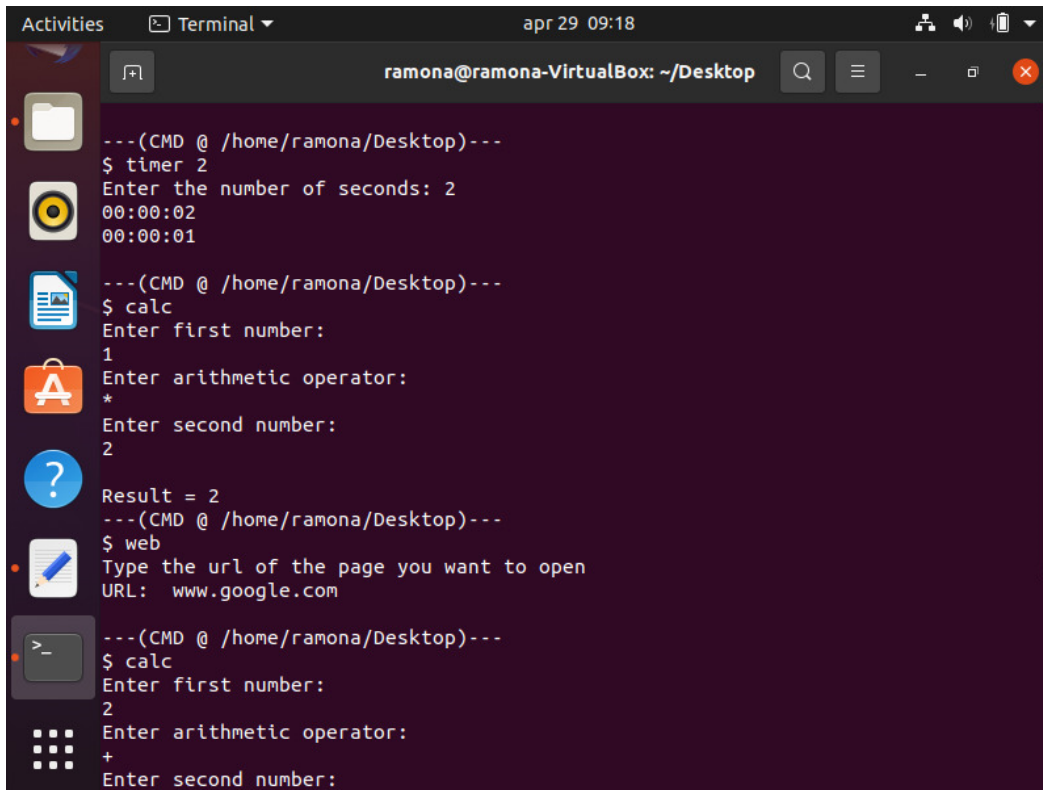
```
24
Enter arithmetic operator:
-
Enter second number:
8
Result = 16
---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
24
Enter arithmetic operator:
*
Enter second number:
8
Result = 192
---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
24
Enter arithmetic operator:
/
Enter second number:
8
Result = 3
---(CMD @ /home/ramona/Desktop)---
$
```



A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" showing the execution of the 'calc' command. The user enters '24' as the first number, '+' as the arithmetic operator, and '8' as the second number. The terminal displays "Result = 32". This process is repeated with '-' as the operator, resulting in "Result = 16".

```
---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
24
Enter arithmetic operator:
+
Enter second number:
8
Result = 32
---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
24
Enter arithmetic operator:
-
Enter second number:
8
Result = 16
---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
24
Enter arithmetic operator:
*
Enter second number:
8
```

4.3 The “history” command



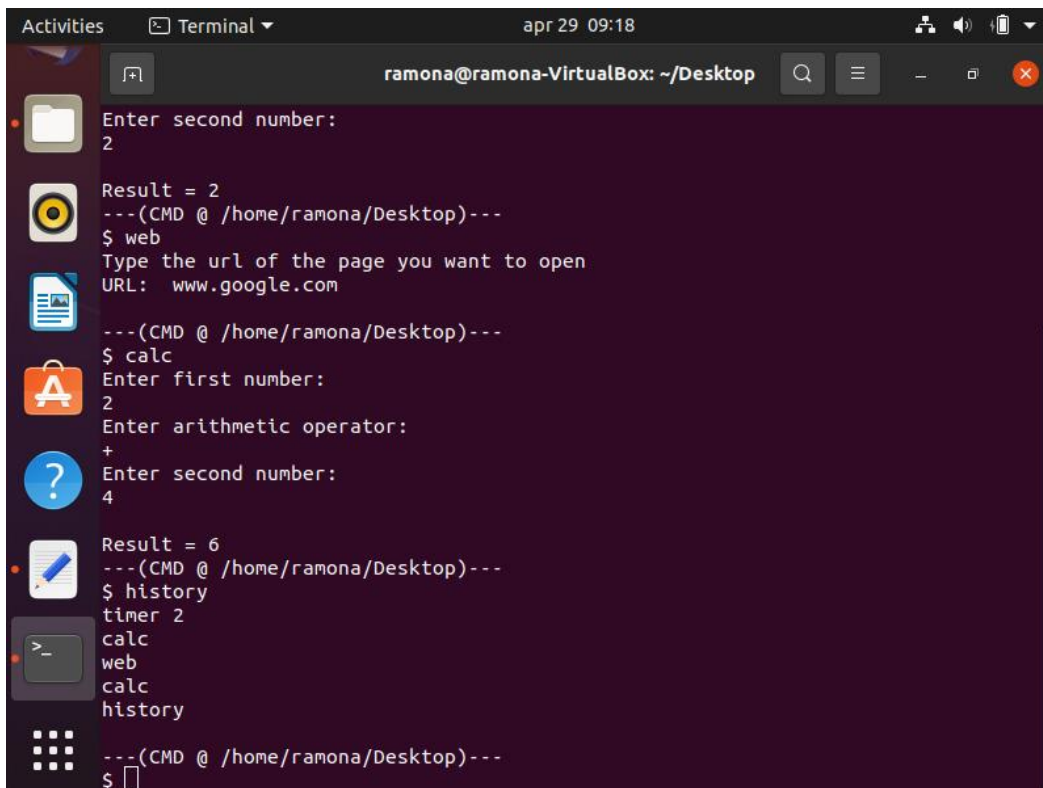
A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" showing the following commands and output:

```
---(CMD @ /home/ramona/Desktop)---
$ timer 2
Enter the number of seconds: 2
00:00:02
00:00:01

---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
1
Enter arithmetic operator:
*
Enter second number:
2
Result = 2

---(CMD @ /home/ramona/Desktop)---
$ web
Type the url of the page you want to open
URL: www.google.com

---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
2
Enter arithmetic operator:
+
Enter second number:
```



A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" showing the continuation of the commands and output from the previous window:

```
Enter second number:
2
Result = 2

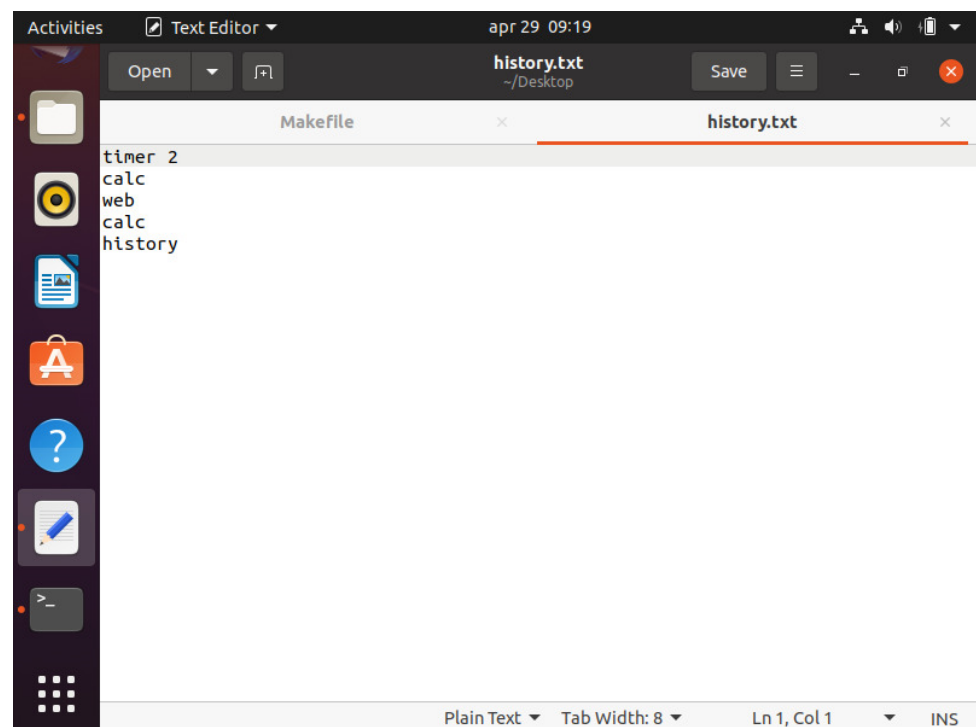
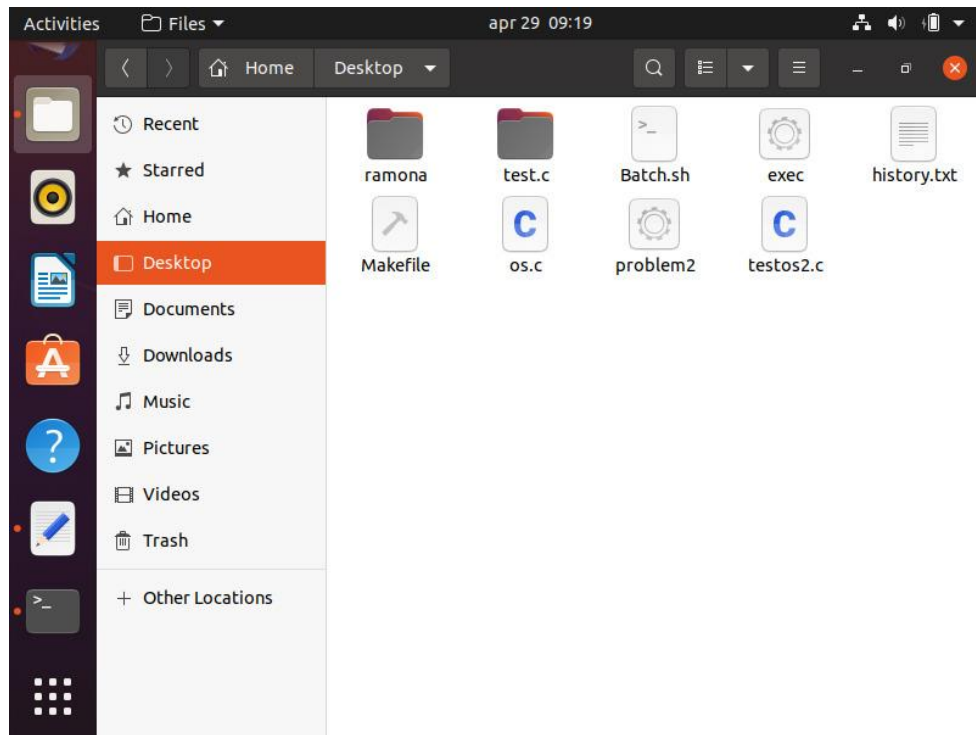
---(CMD @ /home/ramona/Desktop)---
$ web
Type the url of the page you want to open
URL: www.google.com

---(CMD @ /home/ramona/Desktop)---
$ calc
Enter first number:
2
Enter arithmetic operator:
+
Enter second number:
4
Result = 6

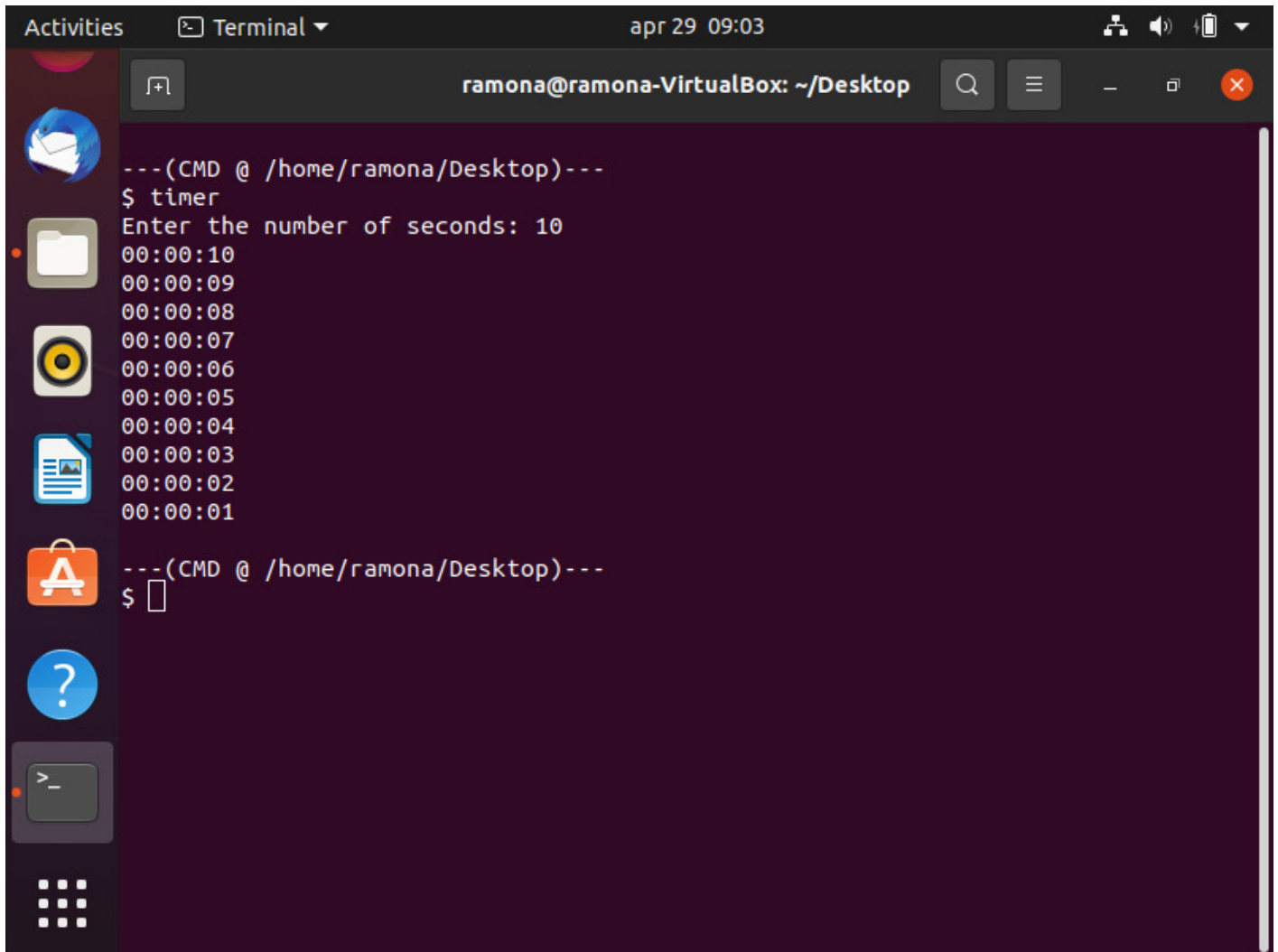
---(CMD @ /home/ramona/Desktop)---
$ history
timer 2
calc
web
calc
history

---(CMD @ /home/ramona/Desktop)---
$
```


- *History command creates a file on the desktop called history where it displays the previous commands used.*



4.4 The “timer” command

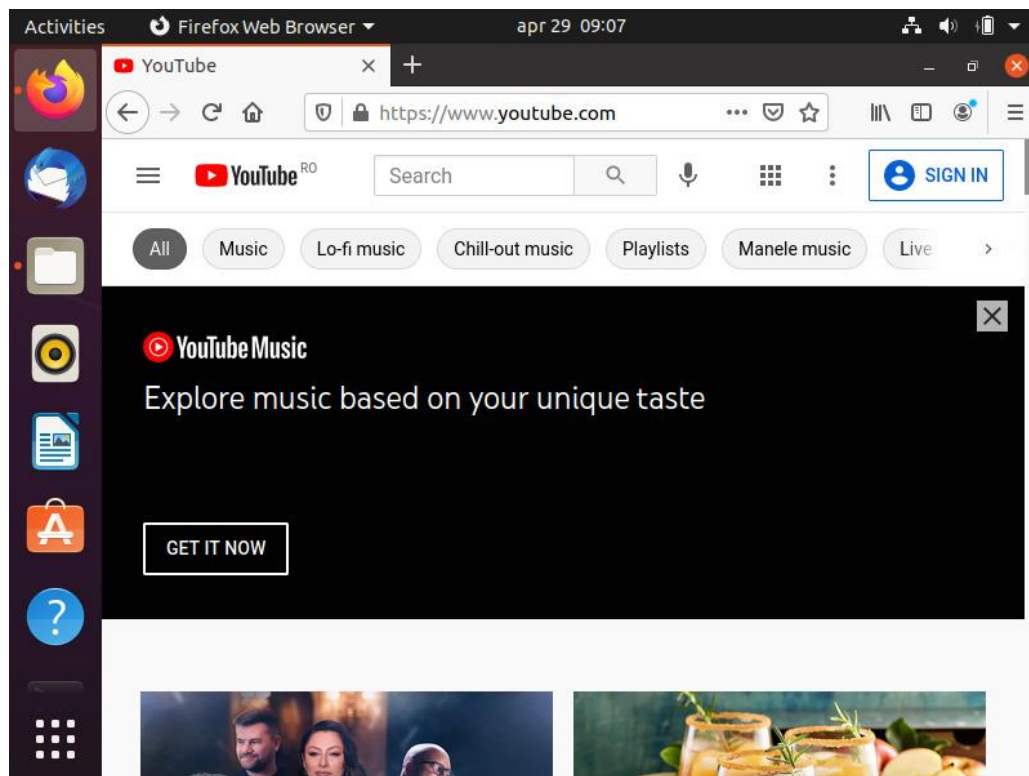
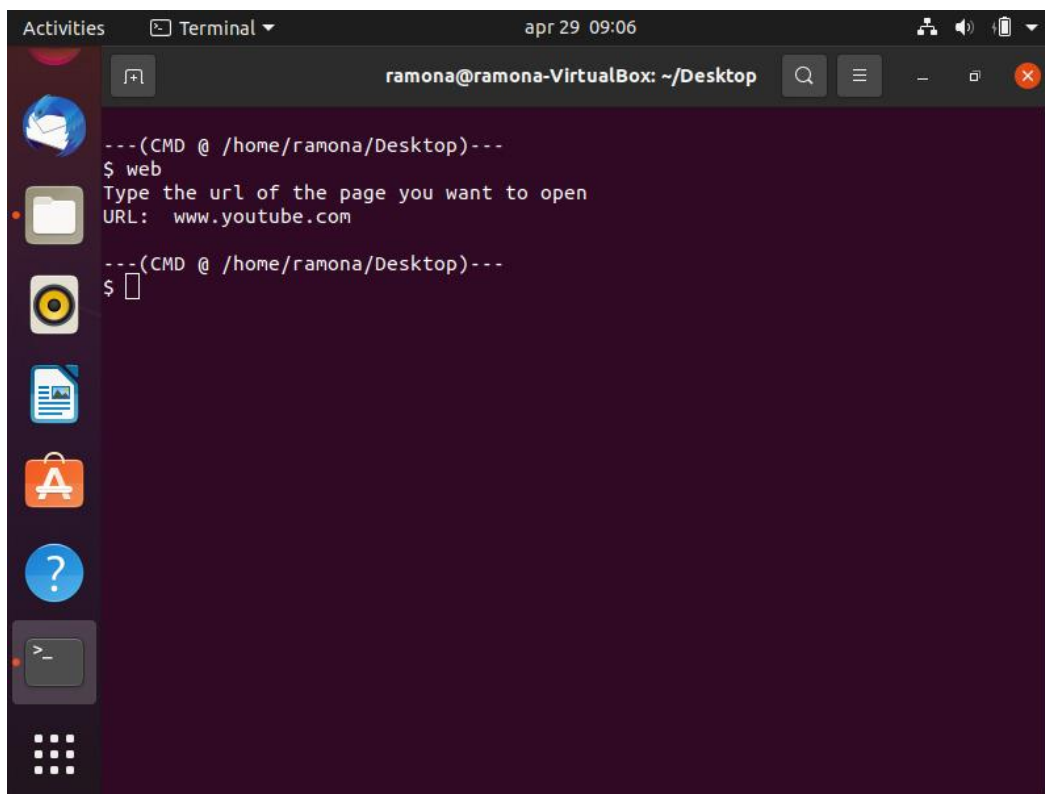


The screenshot shows a terminal window titled "ramona@ramona-VirtualBox: ~/Desktop". The window has a dark theme. On the left side, there is a vertical dock with several application icons: a blue circle with a white envelope, a folder icon, a yellow circle with a black center, a document icon, an orange shopping bag icon, a blue circle with a white question mark, and a terminal icon. The terminal window itself has a dark purple background. The text inside the terminal is as follows:

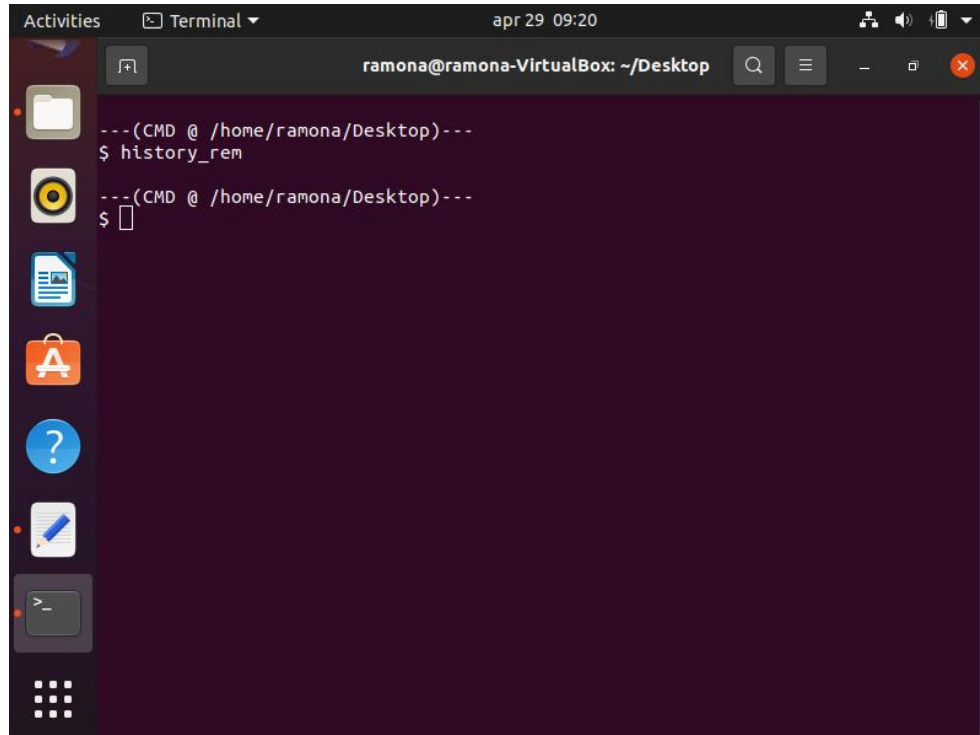
```
---(CMD @ /home/ramona/Desktop)---  
$ timer  
Enter the number of seconds: 10  
00:00:10  
00:00:09  
00:00:08  
00:00:07  
00:00:06  
00:00:05  
00:00:04  
00:00:03  
00:00:02  
00:00:01  
---(CMD @ /home/ramona/Desktop)---  
$
```

The terminal window also has a top bar with "Activities", "Terminal", and the date "apr 29 09:03". On the right side of the top bar, there are icons for network, volume, and battery. Below the top bar, there is a search bar and a menu icon.

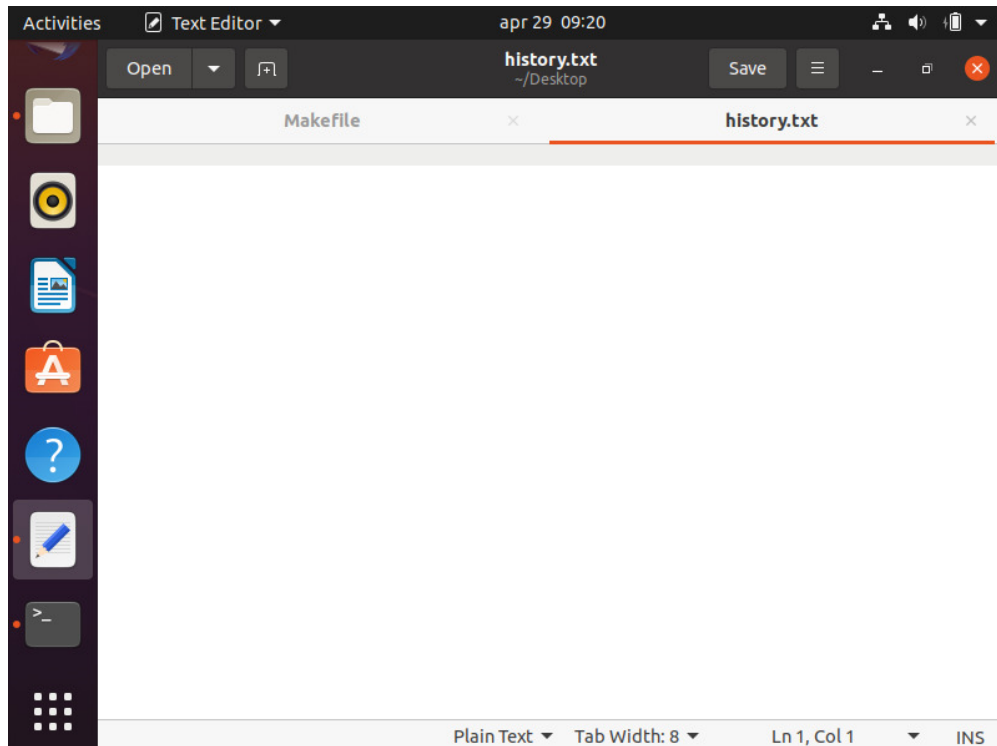
4.5 The “web” command



4.6 The “*history rem*” command



A terminal window titled "ramona@ramona-VirtualBox: ~/Desktop" is shown. The prompt is "---(CMD @ /home/ramona/Desktop)---". The user enters the command `$ history_rem`. The prompt changes to "---(CMD @ /home/ramona/Desktop)---" again, and the user enters a blank line, indicated by a cursor.



A text editor window titled "Text Editor" is shown. The file "history.txt" is open, located at "~/Desktop". The editor shows two tabs: "Makefile" and "history.txt". The "history.txt" tab is active. The editor is in "Plain Text" mode with a "Tab Width" of 8. The cursor is at "Ln 1, Col 1". The status bar at the bottom indicates "INS" mode.

4.7 The “batch” command

```
Activities Terminal apr 26 13:04
ramona@ramona-VirtualBox: ~/Desktop

op/batch.sh)
BATCH: /home
sh: 1: /home: Permission denied

---(CMD @ /home/ramona/Desktop)---
$ batch
Type the thr path and name of the batch you want to execute(ex:/home/user/Desktop/batch.sh)
BATCH: /home/ramona/Desktop/Batch.sh
sh: 1: /home/ramona/Desktop/Batch.sh: Permission denied

---(CMD @ /home/ramona/Desktop)---
$ chmod 777 Batch.sh

---(CMD @ /home/ramona/Desktop)---
$ batch
Type the thr path and name of the batch you want to execute(ex:/home/user/Desktop/batch.sh)
BATCH: /home/ramona/Desktop/Batch.sh
Batch.sh history.txt os.c ramona testos2.c
exec Makefile problem2 test.c
/home/ramona/Desktop
lo no wireless extensions.
enp0s3 no wireless extensions.

---(CMD @ /home/ramona/Desktop)---
$
```

```
Activities Terminal apr 26 13:06
ramona@ramona-VirtualBox: ~/Desktop

---(CMD @ /home/ramona/Desktop)---
$ Batch
Batch: Command not found

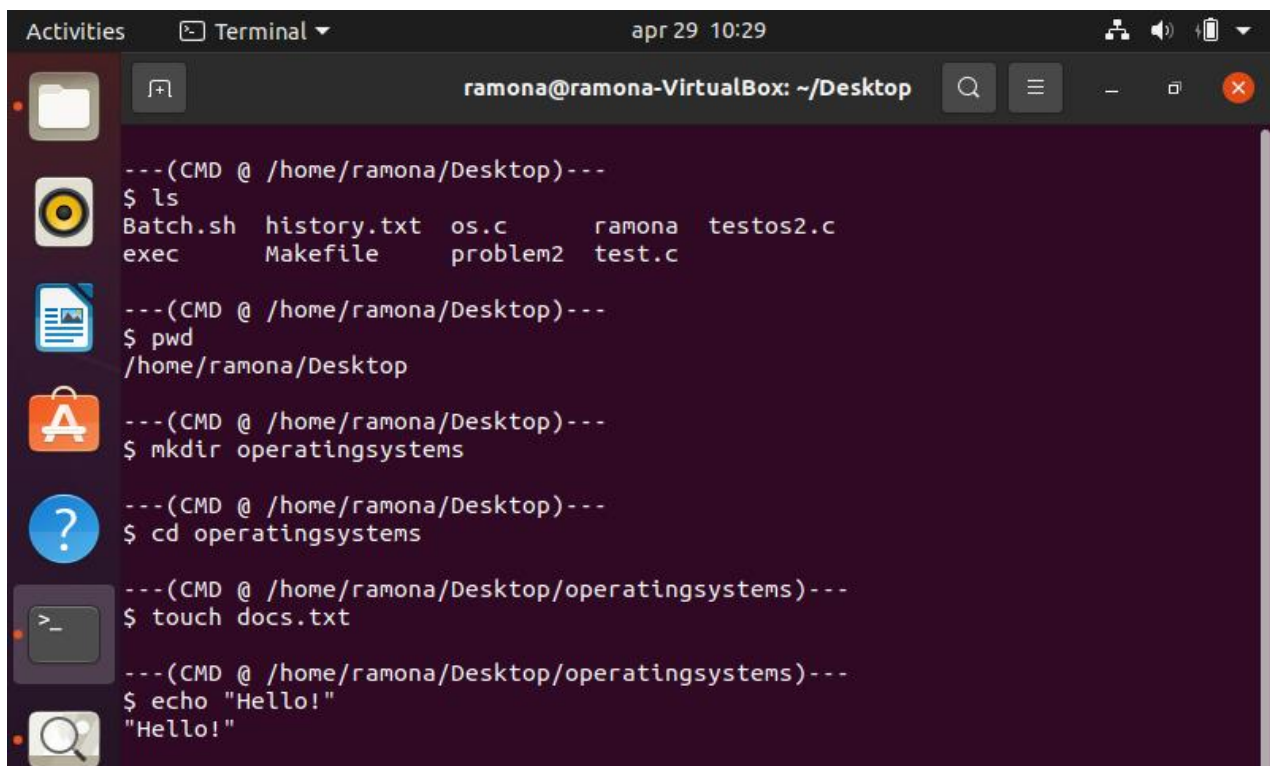
---(CMD @ /home/ramona/Desktop)---
$ batch
Type the thr path and name of the batch you want to execute(ex:/home/user/Desktop/batch.sh)
BATCH: /home
sh: 1: /home: Permission denied

---(CMD @ /home/ramona/Desktop)---
$ batch
Type the thr path and name of the batch you want to execute(ex:/home/user/Desktop/batch.sh)
BATCH: /home/ramona/Desktop/Batch.sh
sh: 1: /home/ramona/Desktop/Batch.sh: Permission denied

---(CMD @ /home/ramona/Desktop)---
$ chmod 777 Batch.sh

---(CMD @ /home/ramona/Desktop)---
$ batch
Type the thr path and name of the batch you want to execute(ex:/home/user/Desktop/batch.sh)
BATCH: /home/ramona/Desktop/Batch.sh
Batch.sh history.txt os.c ramona testos2.c
exec Makefile problem2 test.c
```

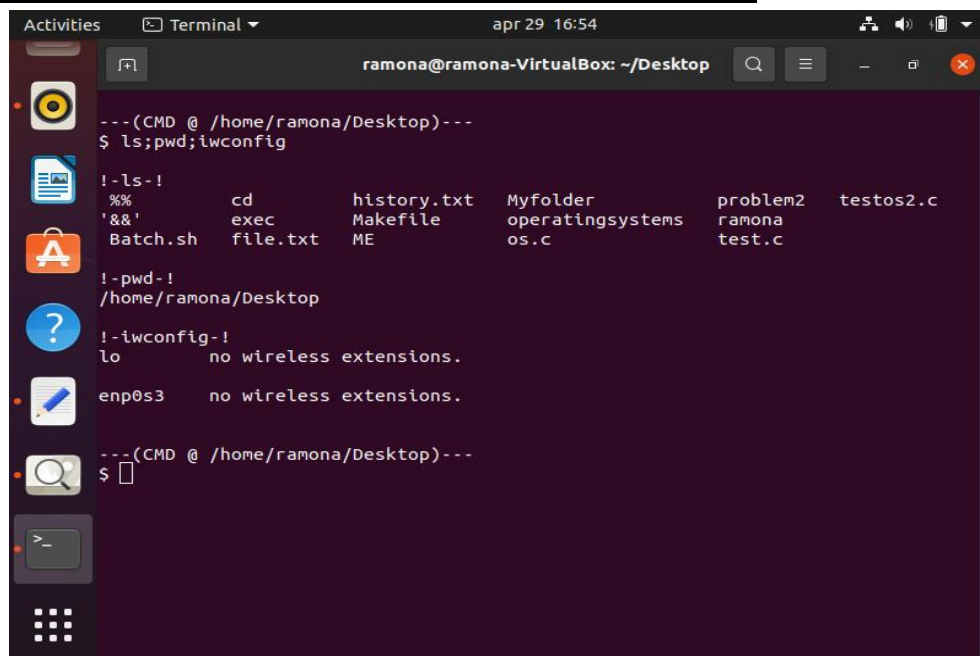
7. Examples of Linux commands:



A screenshot of a Linux terminal window titled "ramona@ramona-VirtualBox: ~/Desktop". The window shows a series of commands and their outputs:

```
---(CMD @ /home/ramona/Desktop)---  
$ ls  
Batch.sh  history.txt  os.c      ramona  testos2.c  
exec      Makefile    problem2  test.c  
  
---(CMD @ /home/ramona/Desktop)---  
$ pwd  
/home/ramona/Desktop  
  
---(CMD @ /home/ramona/Desktop)---  
$ mkdir operatingsystems  
  
---(CMD @ /home/ramona/Desktop)---  
$ cd operatingsystems  
  
---(CMD @ /home/ramona/Desktop/operatingsystems)---  
$ touch docs.txt  
  
---(CMD @ /home/ramona/Desktop/operatingsystems)---  
$ echo "Hello!"  
"Hello!"
```

6. Multiple combined linux commands



A screenshot of a Linux terminal window titled "ramona@ramona-VirtualBox: ~/Desktop". The window shows a series of commands and their outputs:

```
---(CMD @ /home/ramona/Desktop)---  
$ ls;pwd;iwconfig  
  
!-ls-!  
%%      cd      history.txt  Myfolder  problem2  testos2.c  
'&&'    exec     Makefile    operatingsystems  ramona  
Batch.sh file.txt  ME          os.c      test.c  
  
!-pwd-!  
/home/ramona/Desktop  
  
!-iwconfig-!  
lo      no wireless extensions.  
enp0s3  no wireless extensions.  
  
---(CMD @ /home/ramona/Desktop)---  
$
```

Figure 1

```
Activities Terminal apr 29 16:55
ramona@ramona-VirtualBox: ~/Desktop

---(CMD @ /home/ramona/Desktop)---
$ chmod 777 history.txt;echo "This is the history";history

!-chmod 777 history.txt-!

!-echo "This is the history"-!
"This is the history"

!-history-!
clear
help
ls
cd
cd Desktop
clear
ls
pwd
mkdir operatingsystems
cd operatingsystems
rmdir operatingsystems
clear
uptime
clear
echo "Hi there";pwd;uptime;date;
echo "Hi there!";uptime;pwd;
ls;pwd;iwconfig;
mkdir Myfolder && cd Myfolder
mkdir ME %% cd ME
```

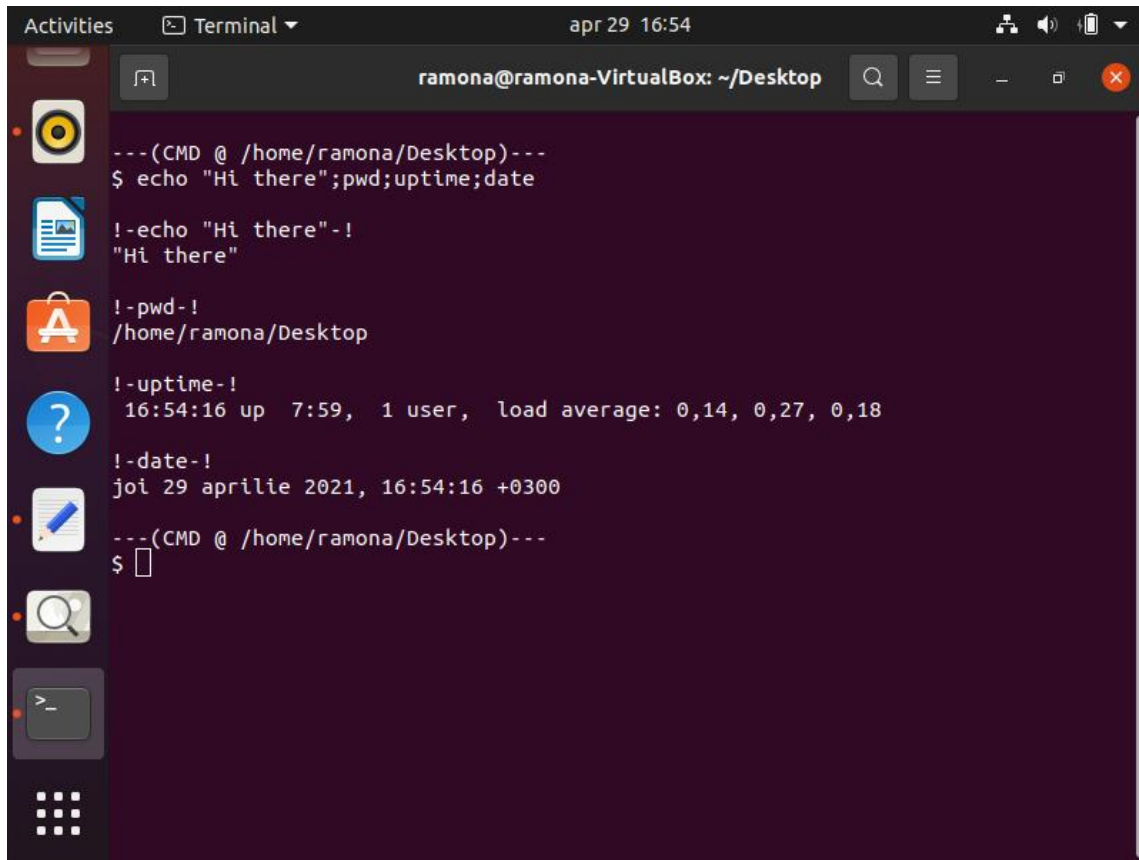
Figure 2.1

```
Activities Terminal apr 29 16:56
ramona@ramona-VirtualBox: ~/Desktop

clear
chmod 777 Desktop;echo "This is the history";history;
echo "Hi there!";pwd;uptime;date;
ls;pwd;iwconfig;ping;
jobs;
ls;pwd;iwconfig;
ls;
help
calc
clar
clear
help
batch
clear
top;ping www.google.com; jobs;
touch abc.txt
clear
touch abc.txt;locate -i abc;pwd;
touch file.txt;pwd;uname;
ls;pwd;iwconfig
clear
echo "Hi there";pwd;uptime;date
clear
ls;pwd;iwconfig
clear
chmod 777 history.txt;echo "This is the history";history

---(CMD @ /home/ramona/Desktop)---
$
```

Figure2.2



The image shows a terminal window titled "ramona@ramona-VirtualBox: ~/Desktop". The window has a dark purple background. On the left side, there is a vertical dock with several application icons: a yellow circle with a black dot, a blue document icon, an orange shopping bag icon, a blue circle with a white question mark, a blue document with a pencil icon, a magnifying glass icon, a terminal icon, and a 3x3 grid of white dots. The terminal text is as follows:

```
---(CMD @ /home/ramona/Desktop)---
$ echo "Hi there";pwd;uptime;date

!-echo "Hi there"-!
"Hi there"

!-pwd-!
/home/ramona/Desktop

!-uptime-!
16:54:16 up 7:59, 1 user, load average: 0,14, 0,27, 0,18

!-date-!
joi 29 aprilie 2021, 16:54:16 +0300

---(CMD @ /home/ramona/Desktop)---
$
```

Figure 3