

Njungle Nges Brian

FE14A169

CEF305 ASSIGEMENT

TITTLE:

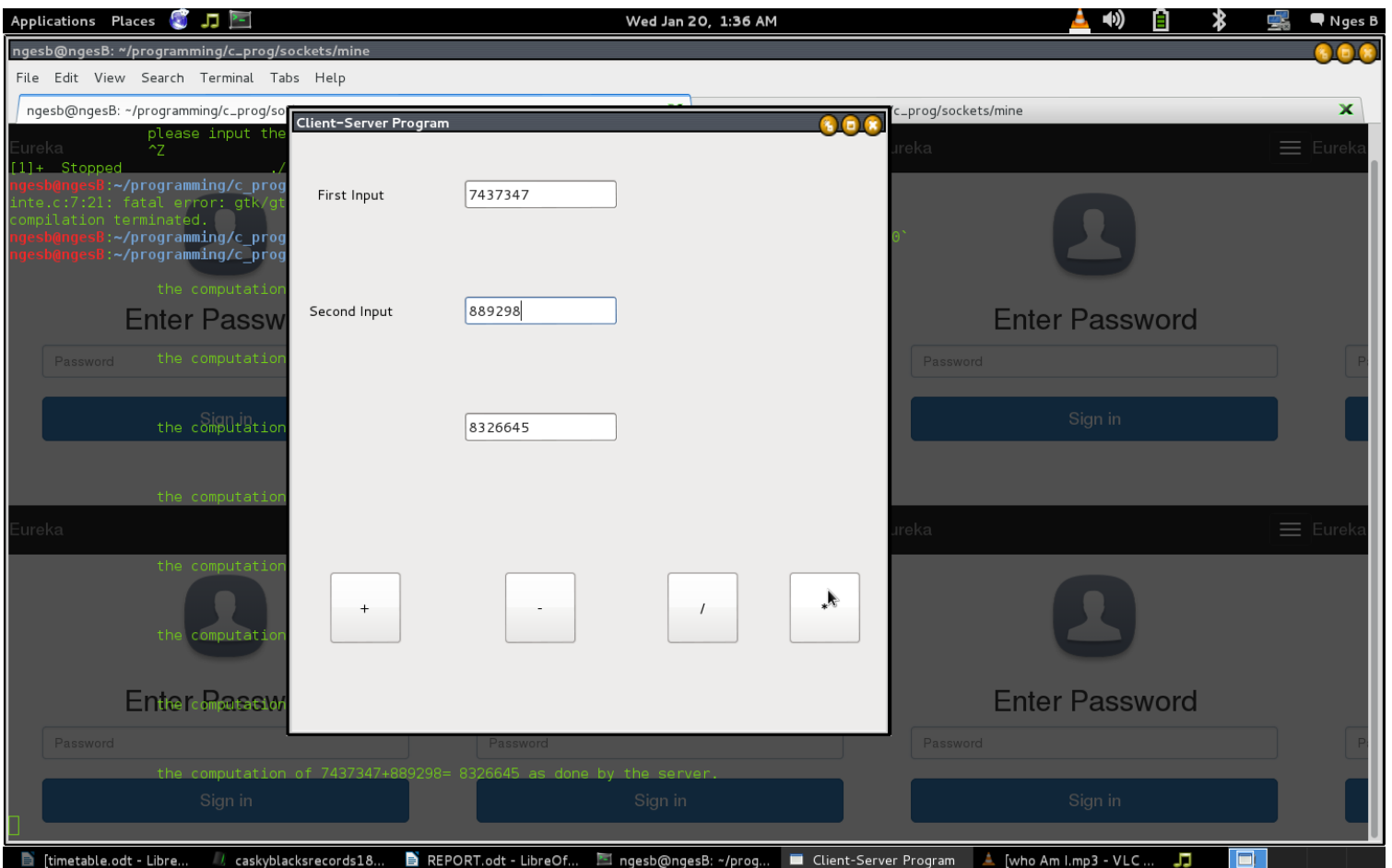
A CLIENT SERVER PROGRAM WITH A GUI DONE IN SOCKET PROGRAMING AND GTK.

PROPERTIES OF MECHINE

Operating system:Kali Linux

Laptop properties :Dell Latitude 6410,Hard drive,320Gb, 5processors,RAM=4Gb

Pictures of the interface



Pictures of Codes

```
Applications  Places  Wed Jan 20, 1:59 AM
server1.c (~/.programming/c-prog/sockets/mine) - gedit
File Edit View Search Tools Documents Help

server1.c x  inte.c x
26 listen(server_sockfd,5);
27 while(1){
28     struct data
29     {
30         int values[2];
31         int result;
32         char operate;
33     }input;
34
35     char wrong[15];
36     strcpy(wrong,"wrong operator");
37
38     printf("server waiting\n");
39
40
41 //accept connection
42
43 client_len=sizeof(client_address);
44 client_sockfd=accept(server_sockfd,(struct sockaddr *)&client_address,&client_len);
45
46 //read and write to client
47 read(client_sockfd,&input,sizeof(input));
48 printf("%c",input.operate);
49 if(input.operate=='+'){
50     input.result=input.values[0]+input.values[1];
51     printf("the computation of %d+%d= %d\n",input.values[0],input.values[1],input.result );
52 }
53 else if(input.operate=='-'){
54     input.result=input.values[0]-input.values[1];
55 }
56 else if(input.operate=='*'){
57     input.result=input.values[0]*input.values[1];
58 }
59 else if (input.operate=='/'){
60     input.result=input.values[0]/input.values[1];
61 }
62 else
63     write(client_sockfd,&wrong,20);
64 write(client_sockfd,&input,sizeof(input));
65 close(client_sockfd);
66 }
67 }
```

```
Applications  Places  Wed Jan 20, 1:57 AM
inte.c (~/.programming/c-prog/sockets/mine) - gedit
File Edit View Search Tools Documents Help

server1.c x  inte.c x
1 #include <stdio.h>
2 #include <sys/types.h>
3 #include <sys/socket.h>
4 #include <sys/un.h>
5 #include <unistd.h>
6 #include <stdlib.h>
7 #include <gtk/gtk.h>
8
9 int Delete(GtkWidget *widget, GtkWidget *event ,gpointer data);
10 void addi( );
11 void sub( );
12 void divi( );
13 void multi( );
14
15
16 struct data
17 {
18     int values[2];
19     int result;
20     char operate;
21 }
```

```
Applications  Places  Wed Jan 20, 1:58 AM
inte.c (~/.programming/c-prog/sockets/mine) - gedit
File Edit View Search Tools Documents Help

server1.c x  inte.c x
44
45 add=gtk_button_new_with_label("+");
46 gtk_widget_set_size_request(add,75,75);
47 sub=gtk_button_new_with_label("-");
48 gtk_widget_set_size_request(sub,75,75);
49 divide=gtk_button_new_with_label("/");
50 gtk_widget_set_size_request(divide,75,75);
51 mul=gtk_button_new_with_label("*");
52 gtk_widget_set_size_request(mul,75,75);
53
54 lab1=gtk_label_new("First Input");
55 lab2=gtk_label_new("Second Input");
56
57 table=gtk_table_new(20,20,FALSE);
58
59 gtk_table_attach(GTK_TABLE(table),lab1,2,5,2,4,1,1,0,0);
60 gtk_table_attach(GTK_TABLE(table),lab2,2,5,5,7,1,1,0,0);
61 gtk_table_attach(GTK_TABLE(table),num1,6,10,2,4,1,1,0,0);
62 gtk_table_attach(GTK_TABLE(table),num2,6,10,5,7,1,1,0,0);
63 gtk_table_attach(GTK_TABLE(table),result,6,10,8,10,1,1,0,0);
64 gtk_table_attach(GTK_TABLE(table),add,2,6,10,14,1,1,0,0);
65 gtk_table_attach(GTK_TABLE(table),sub,6,10,16,14,1,1,0,0);
66 gtk_table_attach(GTK_TABLE(table),divide,10,14,16,14,1,1,0,0);
67 gtk_table_attach(GTK_TABLE(table),mul,14,18,16,14,1,1,0,0);
68
69 char a,s,m,d;
70
71
72 g_signal_connect(G_OBJECT(add),"clicked",G_CALLBACK(addi),NULL);
73 g_signal_connect(G_OBJECT(sub),"clicked",G_CALLBACK(subt),NULL);
74 g_signal_connect(G_OBJECT(divide),"clicked",G_CALLBACK(divi),NULL);
75 g_signal_connect(G_OBJECT(mul),"clicked",G_CALLBACK(multi),NULL);
76
77
78 gtk_container_add(GTK_CONTAINER(window),table);
79 gtk_widget_show_all(window);
80
81 gtk_main();
82 return 0;
83 }
```

Applications Places Wed Jan 20, 1:58 AM

intec.c (~/.programming/c-prog/sockets/mine) - gedit

File Edit View Search Tools Documents Help

server1.c X intec.c X

```
84
85 //the function delete.This function destroy the window when called.
86 int Delete(GtkWidget *widget ,GtkWidget *event ,gpointer data){
87     gtk_main_quit();
88     return FALSE;
89 }
90
91 void addi( ){
92
93
94     int num;
95     char pt[2];
96     input.values[0]=atoi(gtk_entry_get_text(GTK_ENTRY(num1)));
97
98     input.values[1]=atoi(gtk_entry_get_text(GTK_ENTRY(num2)));
99     input.operate='+';
100     //create a socket for the client
101     sockfd=socket(AF_UNIX,SOCK_STREAM,0);
102
103     //name the socket
104     address.sun_family=AF_UNIX;
105     strcpy(address.sun_path,"server_socket");
106     len=sizeof(address);
107
108     //connect our socket to the server
109     con=connect(sockfd,(struct sockaddr *)&address,len);
110     if(con==-1){
111         perror("oops the server could not connect\n");
112         exit(1);
113     }
114
115     write(sockfd, &input,sizeof(input));
116     read(sockfd,&input,sizeof(input));
117     sprintf(pt,"%d",input.result);
118     printf("\n\t\t the computation of %d%c%d= %d as done by the server. \n\n",input.values[0],input.operate,input.values[1],input.result );
119     gtk_entry_set_text(GTK_ENTRY(result),pt);
120     close(sockfd);
121 }
122
123
```

C Tab Width: 4 Ln 3, Col 24 INS

[timetable.odt - Li... [REPORT.odt - Lib... [ngesb@ngesB: ~/... [Client-Server Pro... PEPE mine intec.c (~/.programm...

Applications Places Wed Jan 20, 1:58 AM

intec.c (~/.programming/c-prog/sockets/mine) - gedit

File Edit View Search Tools Documents Help

server1.c X intec.c X

```
121
122 }
123
124 void subtr( ){
125
126
127     int num;
128     char pt[2];
129     input.values[0]=atoi(gtk_entry_get_text(GTK_ENTRY(num1)));
130
131     input.values[1]=atoi(gtk_entry_get_text(GTK_ENTRY(num2)));
132     input.operate='-';
133     //create a socket for the client
134     sockfd=socket(AF_UNIX,SOCK_STREAM,0);
135
136     //name the socket
137     address.sun_family=AF_UNIX;
138     strcpy(address.sun_path,"server_socket");
139     len=sizeof(address);

```

Applications Places Wed Jan 20, 1:59 AM

server1.c (~/.programming/c-prog/sockets/mine) - gedit

File Edit View Search Tools Documents Help

server1.c X intec.c X

```
1 #include <sys/types.h>
2 #include <sys/socket.h>
3 #include <sys/un.h>
4 #include <stdio.h>
5 #include <unistd.h>
6 #include <string.h>
7
8 int main(){
9
10     int server_sockfd,client_sockfd;
11     int server_len,client_len;
12     struct sockaddr_un server_address;
13     struct sockaddr_un client_address;
14
15     //remove any old socket and create a new unnames socket
16     unlink("server_socket");
17     server_sockfd=socket(AF_UNIX,SOCK_STREAM,0);
18
19     //naming the socket
20     server_address.sun_family=AF_UNIX;
21     strcpy(server_address.sun_path,"server_socket");
22     server_len=sizeof(server_address);
23     bind(server_sockfd,(struct sockaddr *)&server_address,server_len);
24
25     //create a connect queue and wait for clients
26     listen(server_sockfd,5);
27     while(1){
28         struct data
29         {
30             int values[2];
31             int result;
32             char operate;
33         }input;
34
35         char wrong[15];
36         strcpy(wrong,"wrong operator");
37
38         printf("server waiting\n");
39
40
```

C Tab Width: 4 Ln 11, Col 31 INS

[timetable.odt - Li... [REPORT.odt - Lib... [ngesb@ngesB: ~/... [Client-Server Pro... PEPE mine server1.c (~/.progr...