1. System calls – mkdir()

System Calls in Operating Systems

System calls are the primary interface between any user application and the operating system kernel. The user programs will request necessary services from kernel by invoking a system call to access hardware devices, manage processes, or interact with the file system. The call is done whenever the user program cannot perform a service because of privilege, and it switches from user mode to kernel mode to execute the service requested.

System Calls are a fundamental aspect of the functional and security aspects of modern operating systems. They control a certain set of interfaces to expose them to system-level resources, thus affecting the overall stability of the system and preventing unauthorized means of manipulating the OS.

mkdir() Function in Linux Lite 7.0

I used the mkdir() system call in a Linux Lite 7.0, and it is used to create a new directory. It is part of the standard C library (<sys/stat.h> and <unistd.h>), providing a programmatic way of organizing a file storage system by dynamically creating directories during the execution of a program.

The function prototype is: int mkdir(name,mode); while,

- name: A string representation of the name(as well as path) of directory to be created
- mode: A permission set, such as read, write, execute, for being formed on the directory, denoted as octal number (e.g.: 0755).

When called, mkdir() attempts to create the specified directory with the permissions indicated. If this operation is a success, it returns 0; otherwise, in failure (examples: no permissions, already existent directory, or improper path), it returns -1 and sets the global errno variable to mark the failure reason.

In Ubuntu-based almost all operating systems such as Linux Lite 7.0, it behaves like any other mkdir() in a normal GNU/Linux environment, and is often found in the context of scripting, software development, and system-level programming where it ensures that directories are created when the system is running without human intervention.

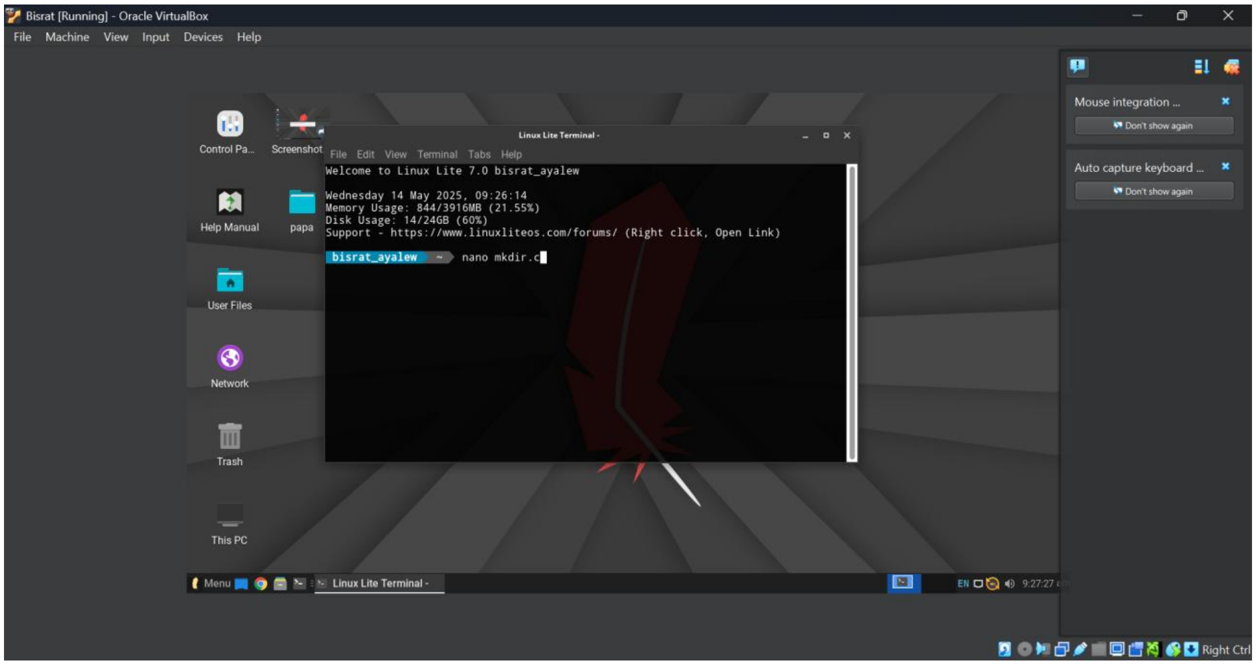
Implementation in C

Here is a simple mkdir() system call implemented in the linux lite terminal that will create a directory named linux after checking if it already exists.

```
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <errno.h>
int main () {
  const char *name = "linux”;
  mode_t type = 0755;
  int result = mkdir(name, type);
  if (result == 0) {
    printf ("successful”);
  } else {
    perror (“failed ");  }
 return 0;}
```

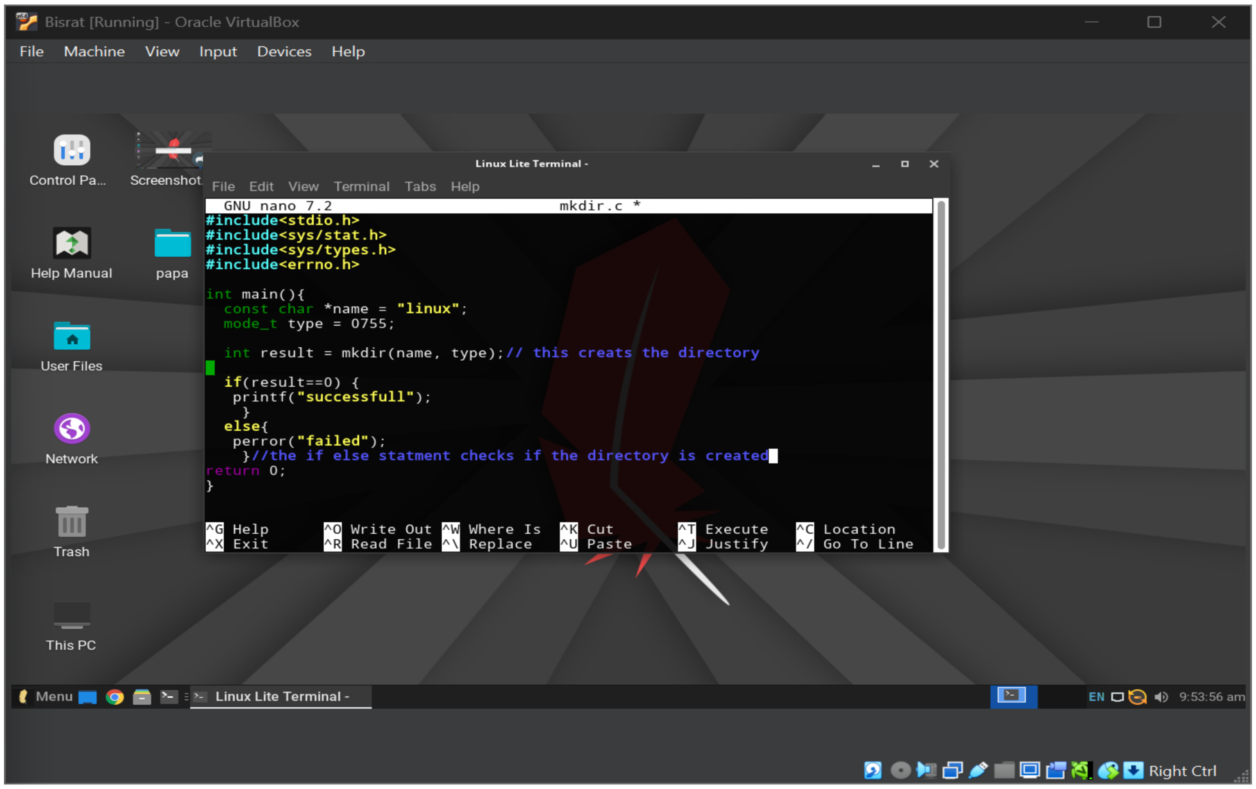
❖ Here are screenshots of the above code implemented in the Linux lite 7.0 terminal

Creating and entering a new c file:



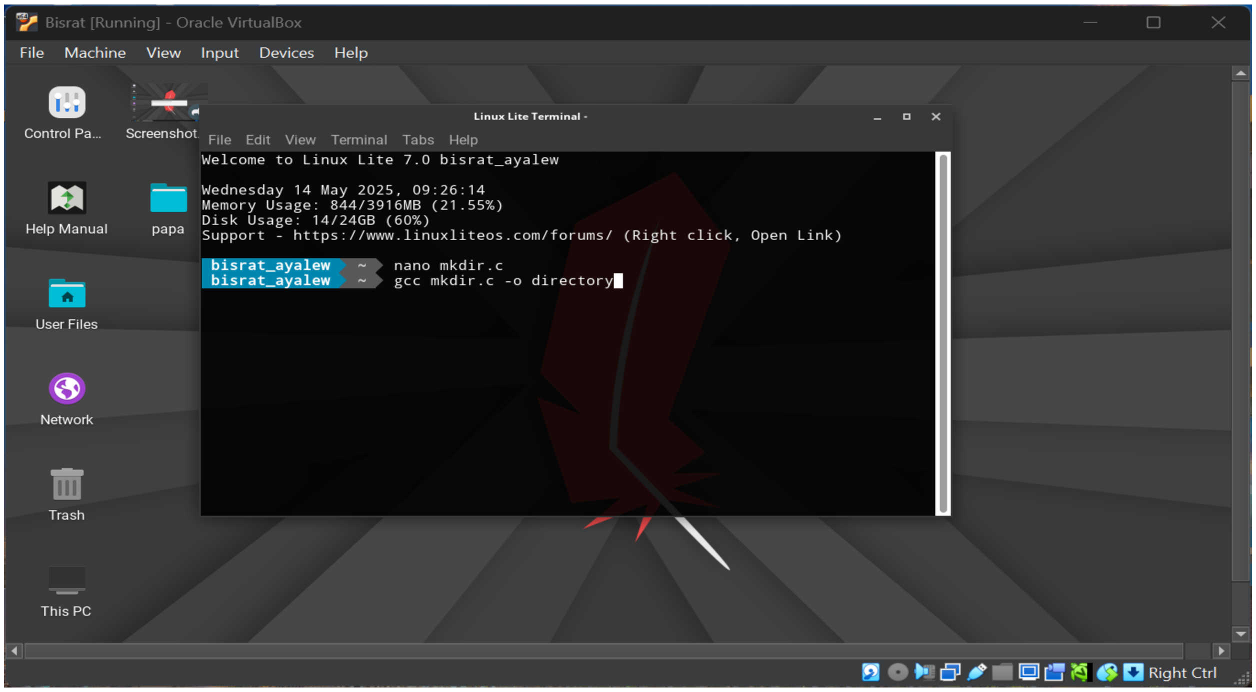
```
Welcome to Linux Lite 7.0 bisrat_ayalew

Wednesday 14 May 2025, 09:26:14
Memory Usage: 844/3916MB (21.55%)
Disk Usage: 14/24GB (60%)
Support - https://www.linuxliteos.com/forums/ (Right click, Open Link)

 bisrat_ayalew  ~  nano mkdir.c
```

Writing the code:

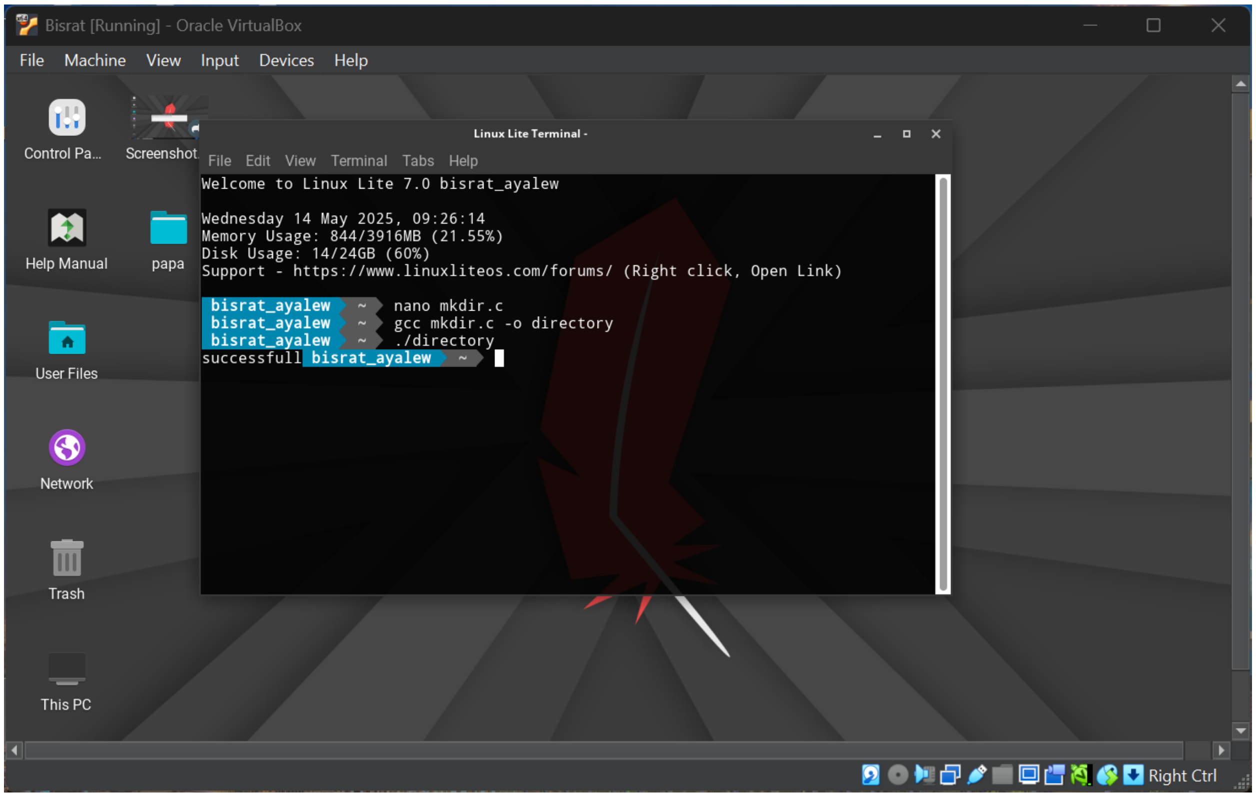


```
  GNU nano 7.2                    mkdir.c *
#include<stdio.h>
#include<sys/stat.h>
#include<sys/types.h>
#include<errno.h>

int main(){
  const char *name = "linux";
  mode_t type = 0755;

  int result = mkdir(name, type);// this creats the directory

  if(result==0) {
   printf("successfull");
    }
  else{
   perror("failed");
    }//the if else statment checks if the directory is created
return 0;
}

^G Help       ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line
```

Compiling and executing:

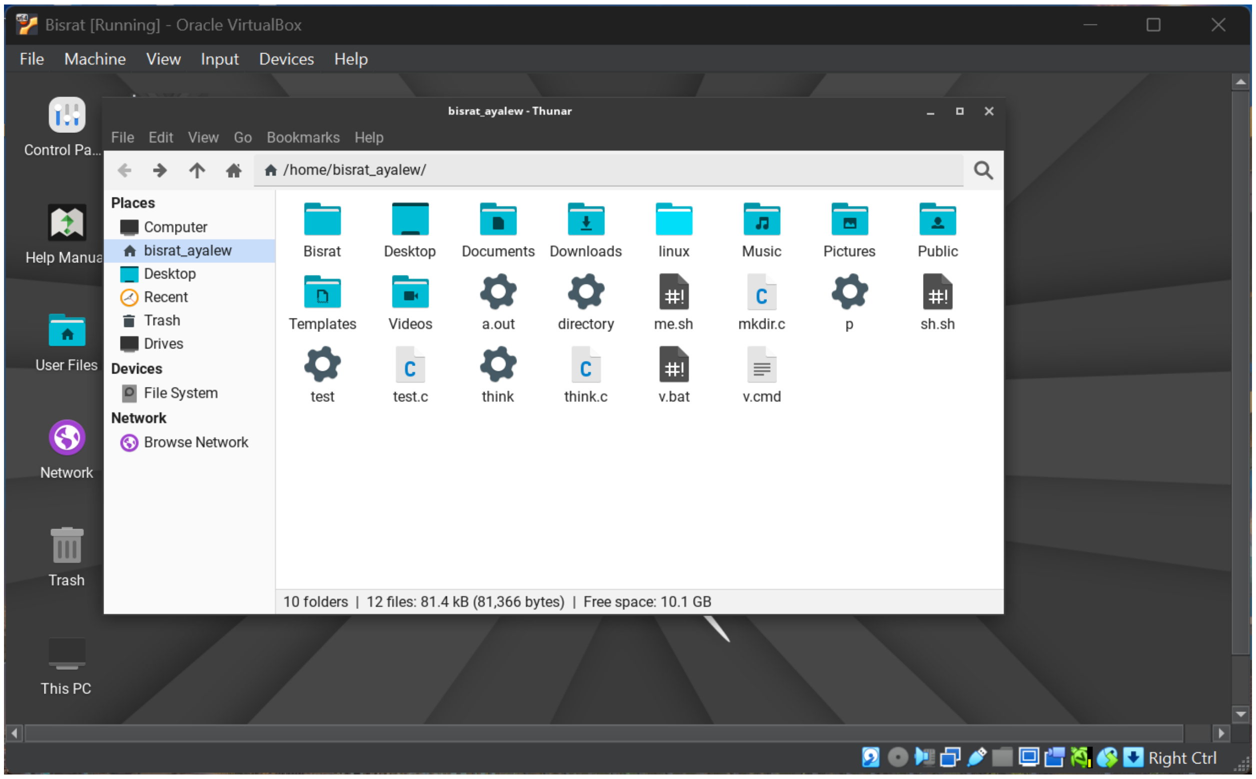

Bisrat [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Control Pa...
Screenshot
Help Manual
papa
User Files
Network
Trash
This PC
Linux Lite Terminal -
File Edit View Terminal Tabs Help
Right Ctrl

```
Welcome to Linux Lite 7.0 bisrat_ayalew

Wednesday 14 May 2025, 09:26:14
Memory Usage: 844/3916MB (21.55%)
Disk Usage: 14/24GB (60%)
Support - https://www.linuxliteos.com/forums/ (Right click, Open Link)

 bisrat_ayalew  ~  nano mkdir.c
 bisrat_ayalew  ~  gcc mkdir.c -o directory
```

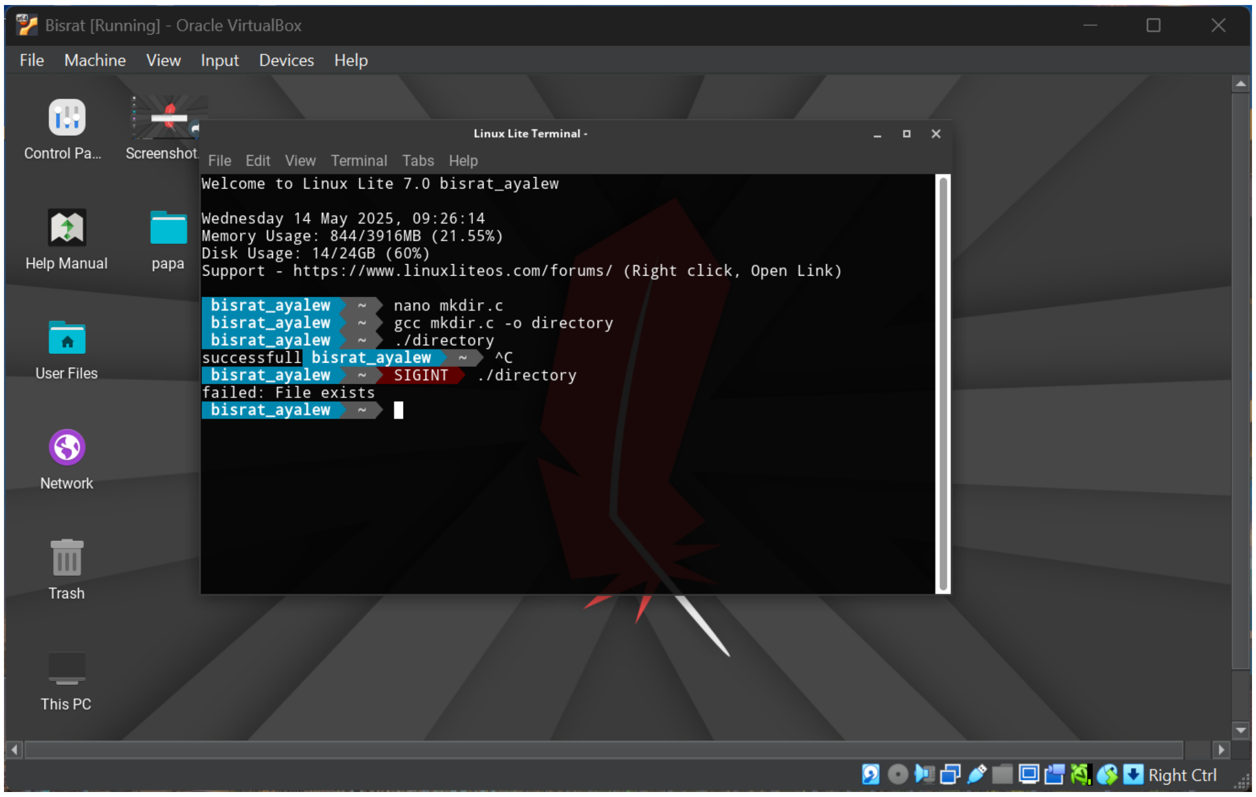

Bisrat [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Control Pa...
Screenshot
Help Manual
papa
User Files
Network
Trash
This PC
Linux Lite Terminal -
File Edit View Terminal Tabs Help
Right Ctrl

```
Welcome to Linux Lite 7.0 bisrat_ayalew

Wednesday 14 May 2025, 09:26:14
Memory Usage: 844/3916MB (21.55%)
Disk Usage: 14/24GB (60%)
Support - https://www.linuxliteos.com/forums/ (Right click, Open Link)

 bisrat_ayalew  ~  nano mkdir.c
 bisrat_ayalew  ~  gcc mkdir.c -o directory
 bisrat_ayalew  ~  ./directory
successfull bisrat_ayalew  ~
```

Here in bisrat_ayalew we can see that the linux directory is created successfully:



When I attempt to create the same directory for the second time, you can see that it says failed after checking the if-else statement I listed above because the file already exists.


Bisrat [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Control Pa...
Screenshot
Help Manual
papa
User Files
Network
Trash
This PC
Linux Lite Terminal -
File Edit View Terminal Tabs Help
Right Ctrl

```
Welcome to Linux Lite 7.0 bisrat_ayalew

Wednesday 14 May 2025, 09:26:14
Memory Usage: 844/3916MB (21.55%)
Disk Usage: 14/24GB (60%)
Support - https://www.linuxliteos.com/forums/ (Right click, Open Link)

 bisrat_ayalew  ~  nano mkdir.c
 bisrat_ayalew  ~  gcc mkdir.c -o directory
 bisrat_ayalew  ~  ./directory
successfull bisrat_ayalew  ~  ^C
 bisrat_ayalew  ~  SIGINT  ./directory
failed: File exists
 bisrat_ayalew  ~ 
```