CS3003D
Operating Systems
ASSIGNMENT 2

AADHARSH K XAVIER
B200730CS

Problem Statement

1. Create a loadable kernel module BINARY SEARCH TREE where you can take the input values as 11, 6, 8, 19, 4, 10, 5, 17, 43, 49, 31 and perform the insertion operation using the C programming in the Linux operating systems.

2. Then remove the created kernel module BINARY SEARCH TREE from the existing list of modules.

Procedure:

1. Save the Binary search tree code as BST.c.
2. Create a Makefile.

```
obj-m += BST.o
all:
	make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
clean:
	make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

3. Execute the make command in this directory. This creates a number of output files including the **BST.ko** which is the kernel module to be loaded.

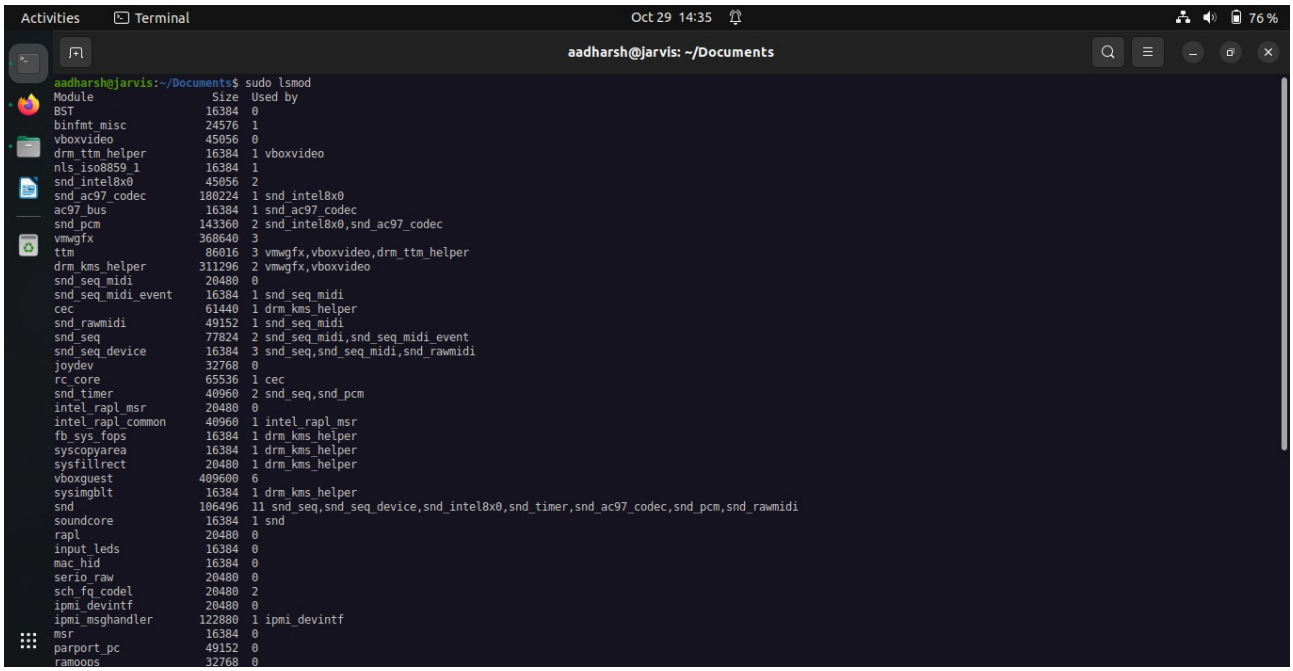
4. Load the kernel module using the insmod command.

```
sudo insmod BST.ko
```

5. List the kernel modules currently loaded in the machine using the lsmod command.

```
sudo lsmod
```

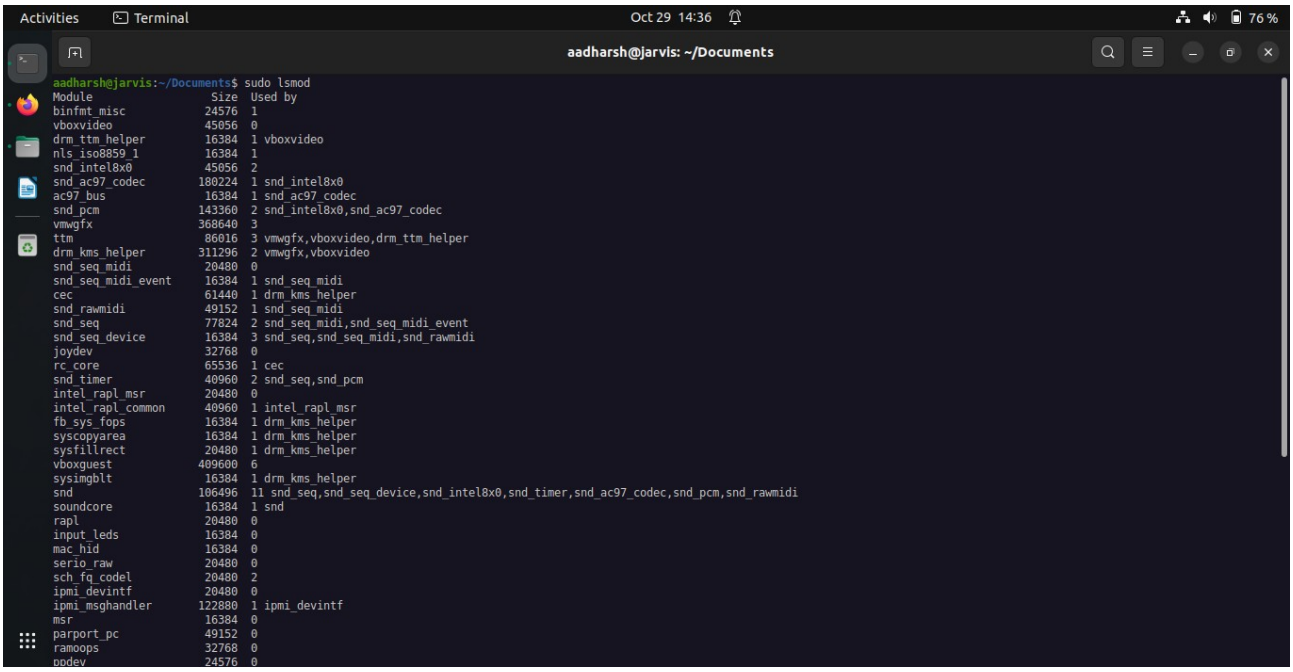
The newly loaded kernel module BST is visible with the other modules.



6. Remove the kernel module using the rmmod command.

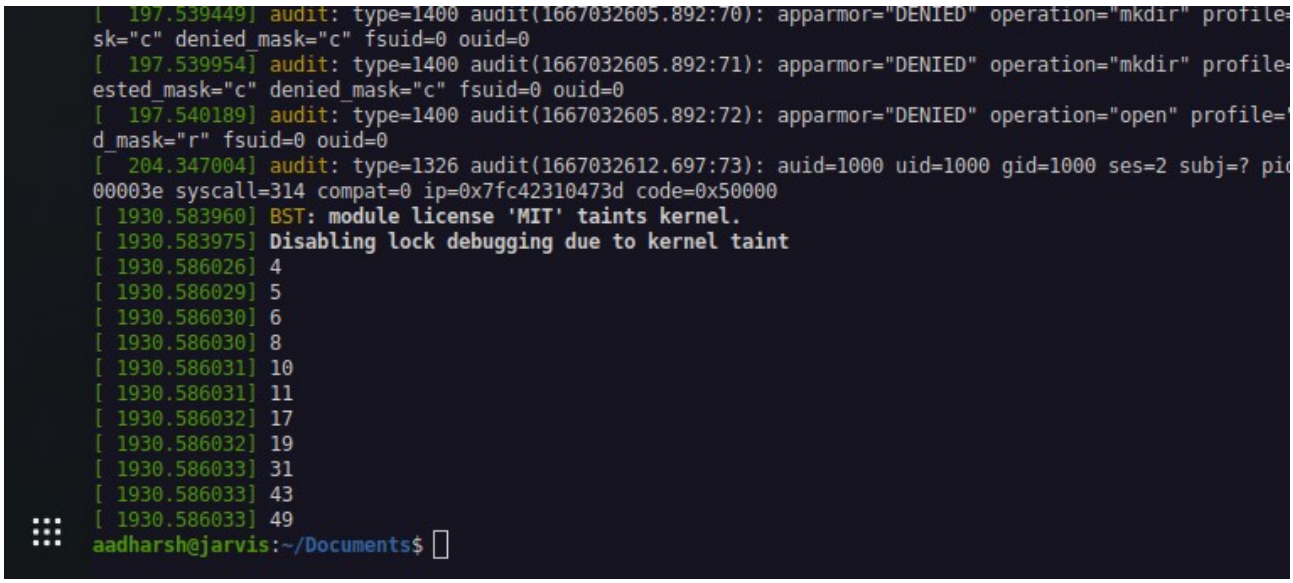
```
sudo rmmod BST
```

7. When the lsmod command is executed again, BST module is no longer present as the module has been successfully removed.



8. The inorder traversal of the Binary search tree after insertion is printed into the kernel log buffer. To confirm whether the operation was successful, execute the dmesg command which displays the contents of the kernel log buffer.

```
sudo dmesg
```



9. Inorder traversal has been printed correctly which implies module creation was successful.