EXPERIMENT 9

NAME: Rahil Sharma
PRN: 18070123062
BATCH: EA-3
SUBJECT: ESRTOS

AIM: Write a C program to output Thread ID in C on Linux

THEORY: In a Unix/Linux operating system, the C/C++ languages provide the POSIX thread(pthread) standard API(Application program Interface) for all thread related functions. It allows us to create multiple threads for concurrent process flow.

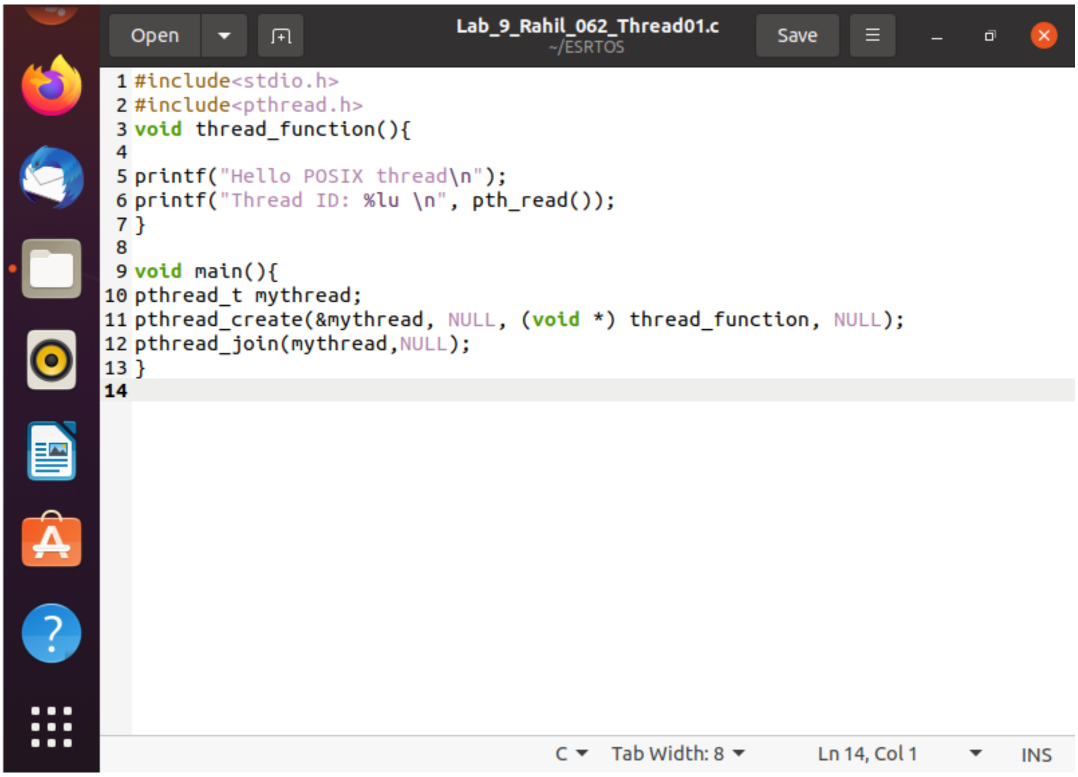
CODES AND OUTPUT OF THE PROGRAM:

i. Thread ID:

Code of the Program:

```
#include<stdio.h>
#include<pthread.h>
void thread_function(){
printf("Hello POSIX thread\n");
printf("Thread ID: %lu \n", pthread_self());
}
void main(){
pthread_t mythread;
pthread_create(&mythread, NULL, (void *) thread_function, NULL);
pthread_join(mythread, NULL);
}
```

Screenshot of the Code:



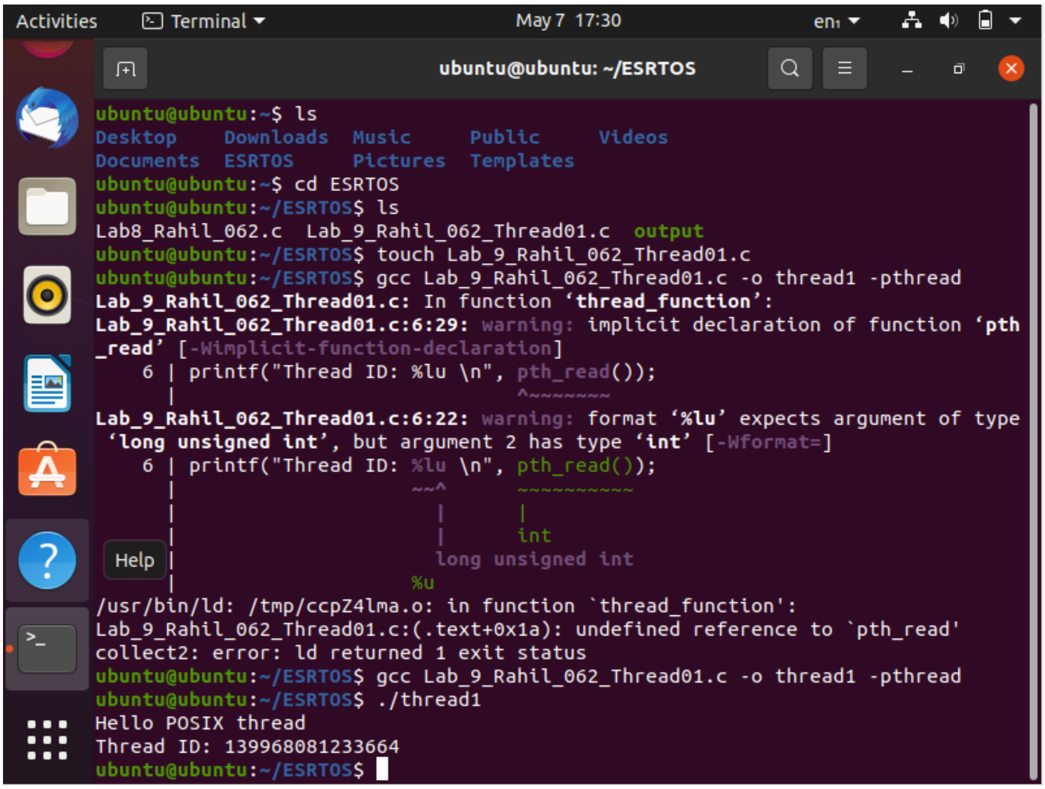
```c
#include<stdio.h>
#include<pthread.h>
void thread_function(){

printf("Hello POSIX thread\n");
printf("Thread ID: %lu \n", pth_read());
}

void main(){
pthread_t mythread;
pthread_create(&mythread, NULL, (void *) thread_function, NULL);
pthread_join(mythread,NULL);
}

```

Screenshot of the Output:



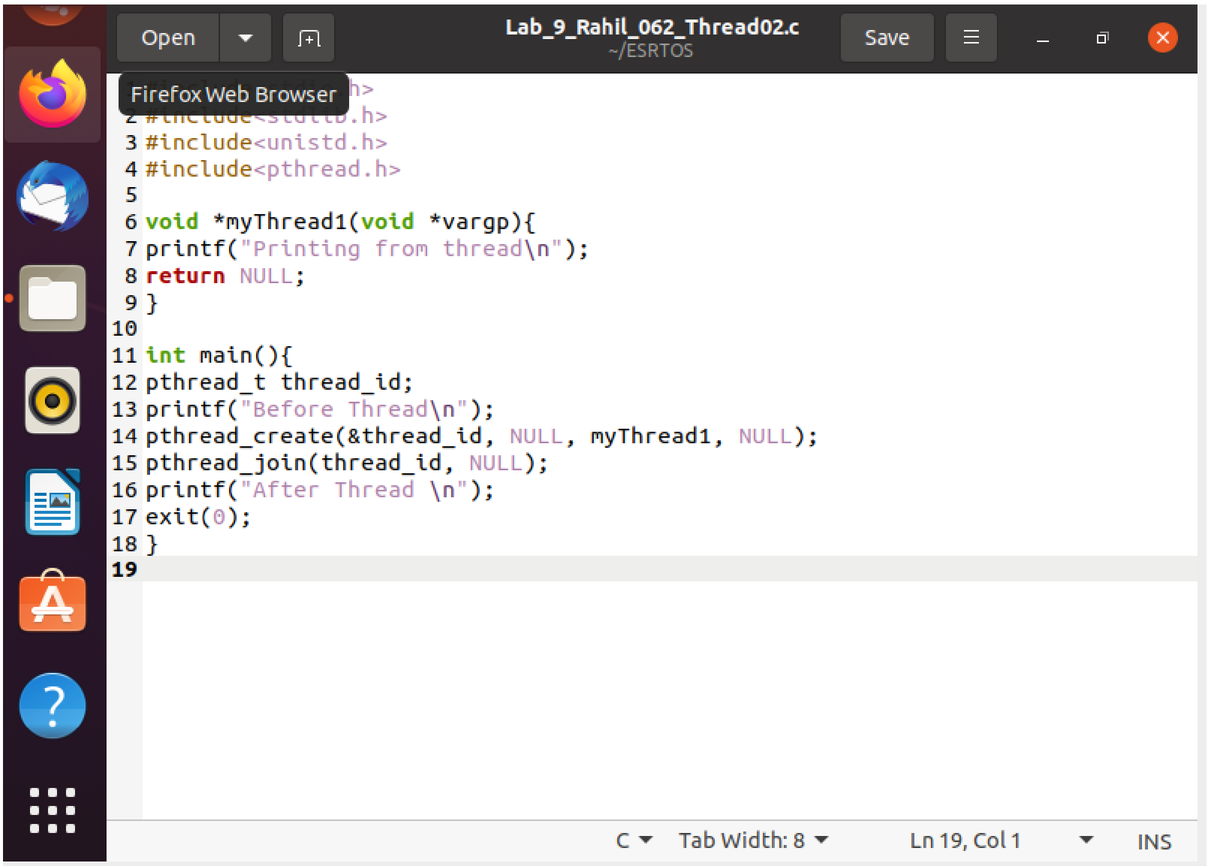
```
ubuntu@ubuntu:~$ ls
Desktop    Downloads  Music     Public     Videos
Documents  ESRTOS     Pictures  Templates
ubuntu@ubuntu:~$ cd ESRTOS
ubuntu@ubuntu:~/ESRTOS$ ls
Lab8_Rahil_062.c  Lab_9_Rahil_062_Thread01.c  output
ubuntu@ubuntu:~/ESRTOS$ touch Lab_9_Rahil_062_Thread01.c
ubuntu@ubuntu:~/ESRTOS$ gcc Lab_9_Rahil_062_Thread01.c -o thread1 -pthread
Lab_9_Rahil_062_Thread01.c: In function ‘thread_function’:
Lab_9_Rahil_062_Thread01.c:6:29: warning: implicit declaration of function ‘pth
_read’ [-Wimplicit-function-declaration]
    6 | printf("Thread ID: %lu \n", pth_read());
      |                             ^~~~~~~~
Lab_9_Rahil_062_Thread01.c:6:22: warning: format ‘%lu’ expects argument of type
 ‘long unsigned int’, but argument 2 has type ‘int’ [-Wformat=]
    6 | printf("Thread ID: %lu \n", pth_read());
      |                    ~~^      ~~~~~~~~~~
      |                      |      |
      |                      |      int
      |                      long unsigned int
      |                    %u
/usr/bin/ld: /tmp/ccpZ4lma.o: in function `thread_function':
Lab_9_Rahil_062_Thread01.c:(.text+0x1a): undefined reference to `pth_read'
collect2: error: ld returned 1 exit status
ubuntu@ubuntu:~/ESRTOS$ gcc Lab_9_Rahil_062_Thread01.c -o thread1 -pthread
ubuntu@ubuntu:~/ESRTOS$ ./thread1
Hello POSIX thread
Thread ID: 139968081233664
ubuntu@ubuntu:~/ESRTOS$
```

ii. Part 2 C Program

Code of the Program:

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<pthread.h>
void *myThread1(void *vargp){
printf("Printing from thread\n");
return NULL;
}
int main(){
pthread_t thread_id;
printf("Before Thread\n");
pthread_create(&thread_id, NULL, myThread1, NULL);
pthread_join(thread_id, NULL);
printf("After Thread \n");
exit(0);
}
```

Screenshot of the Code:

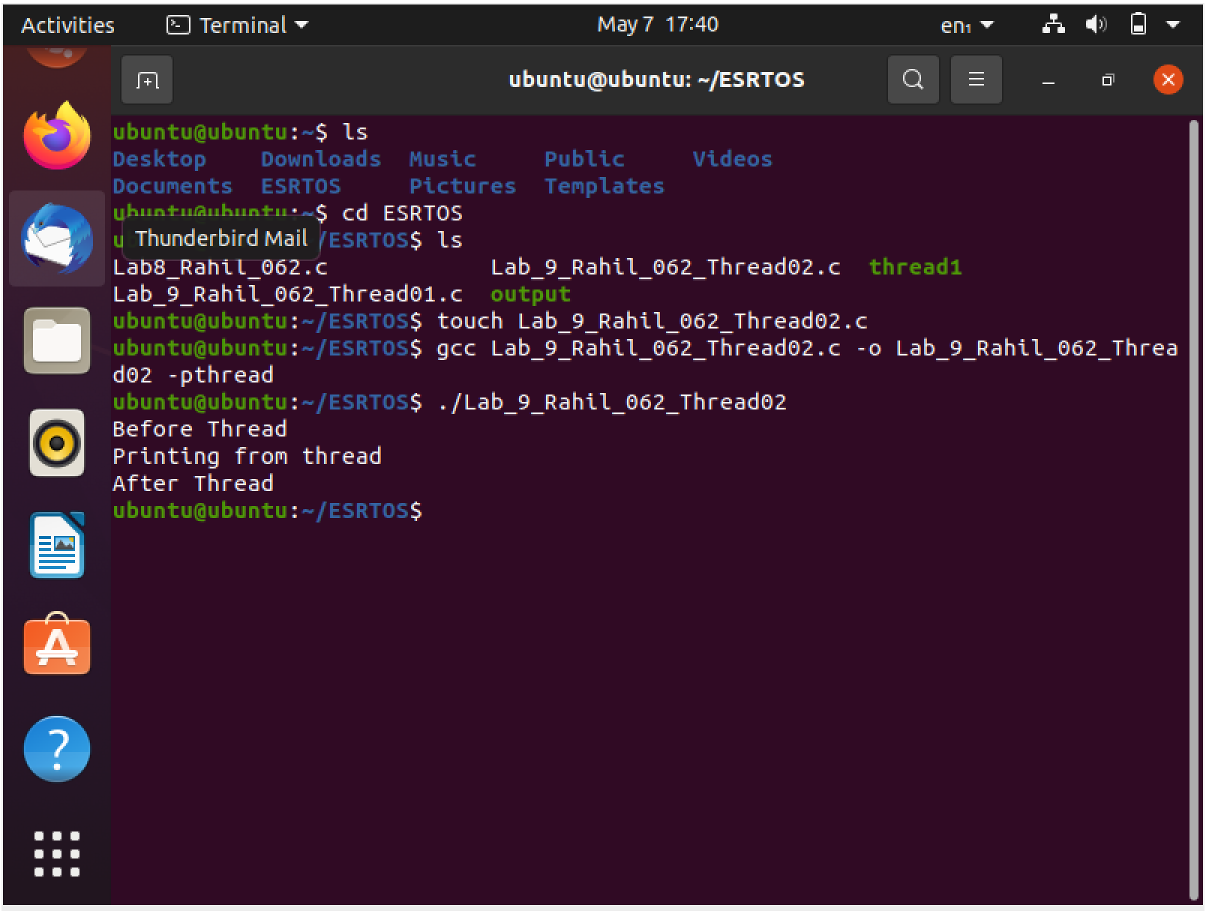


```
h>
#include<stdlib.h>
#include<unistd.h>
#include<pthread.h>

void *myThread1(void *vargp){
printf("Printing from thread\n");
return NULL;
}

int main(){
pthread_t thread_id;
printf("Before Thread\n");
pthread_create(&thread_id, NULL, myThread1, NULL);
pthread_join(thread_id, NULL);
printf("After Thread \n");
exit(0);
}
```

Screenshot of the Output:



```
ubuntu@ubuntu:~$ ls
Desktop    Downloads  Music     Public     Videos
Documents  ESRTOS     Pictures  Templates
ubuntu@ubuntu:~$ cd ESRTOS
ubuntu@ubuntu:~/ESRTOS$ ls
Lab8_Rahil_062.c            Lab_9_Rahil_062_Thread02.c  thread1
Lab_9_Rahil_062_Thread01.c  output
ubuntu@ubuntu:~/ESRTOS$ touch Lab_9_Rahil_062_Thread02.c
ubuntu@ubuntu:~/ESRTOS$ gcc Lab_9_Rahil_062_Thread02.c -o Lab_9_Rahil_062_Threa
d02 -pthread
ubuntu@ubuntu:~/ESRTOS$ ./Lab_9_Rahil_062_Thread02
Before Thread
Printing from thread
After Thread
ubuntu@ubuntu:~/ESRTOS$
```

CONCLUSION: From this experiment we have learnt about the concept of Threading in C and Linux.