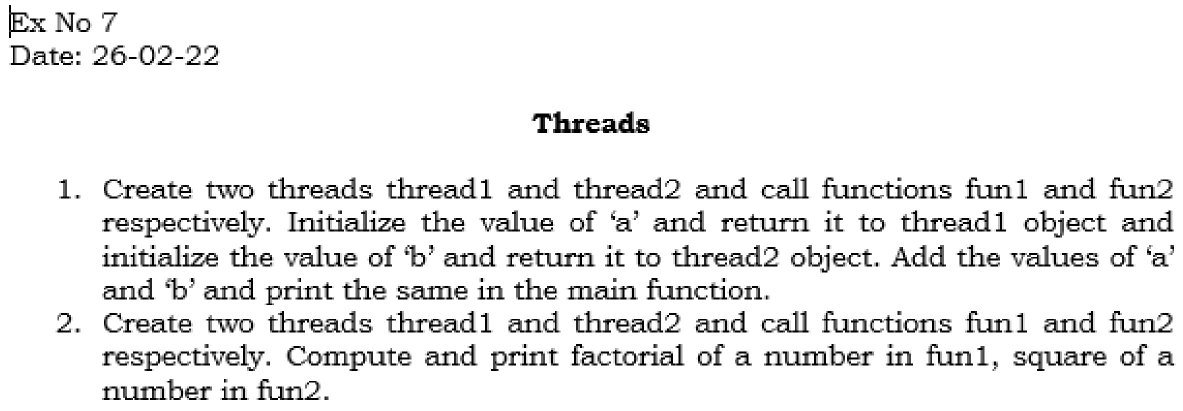
Naming the file: ANSHGOEL_20BCE1798_EX-7_THREADS

RegNo:20BCE1798
Name: ANSH GOEL

Course Code: CSE2005
Course Name: Operating Systems (Embedded Lab)

Slot: L27+L28

Ex No 7: Basic Commands
Date: 26-02-22



Ex No 7
Date: 26-02-22

Threads

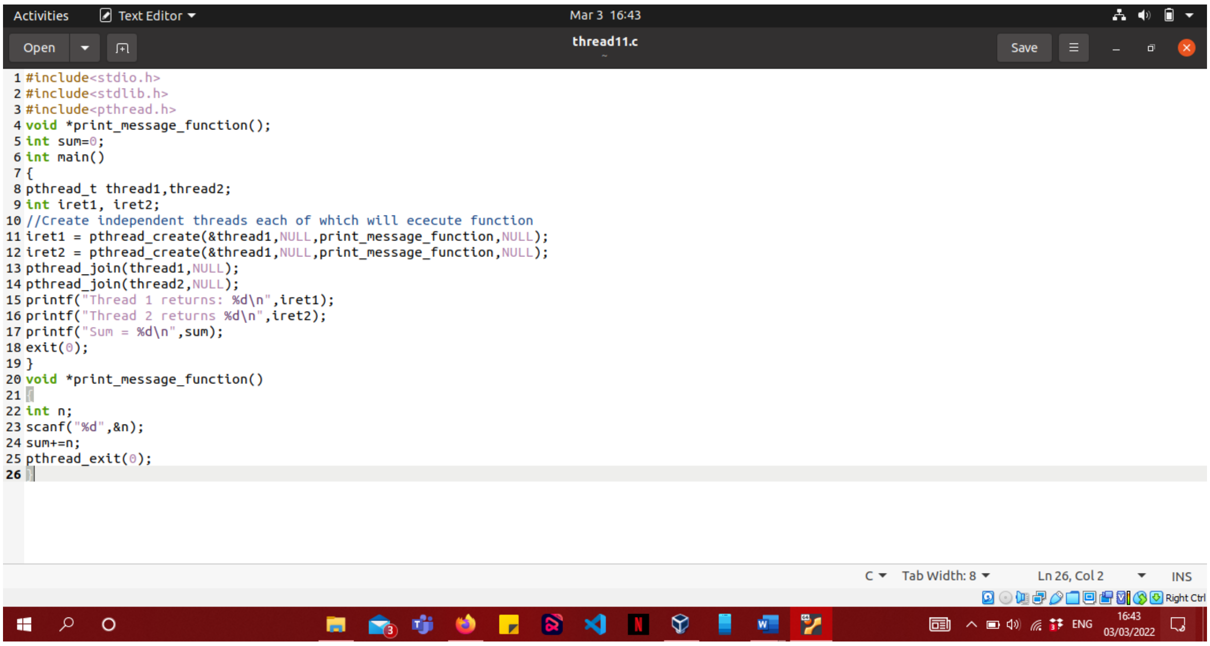
1. Create two threads thread1 and thread2 and call functions fun1 and fun2 respectively. Initialize the value of 'a' and return it to thread1 object and initialize the value of 'b' and return it to thread2 object. Add the values of 'a' and 'b' and print the same in the main function.
2. Create two threads thread1 and thread2 and call functions fun1 and fun2 respectively. Compute and print factorial of a number in fun1, square of a number in fun2.

Q.1.

Create two threads thread1 and thread2 and call functions fun1 and fun2 respectively. Initialize the value of 'a' and return it to thread1 object and initialize the value of 'b' and return it to thread2 object. Add the values of 'a' and 'b' and print the same in the main function.

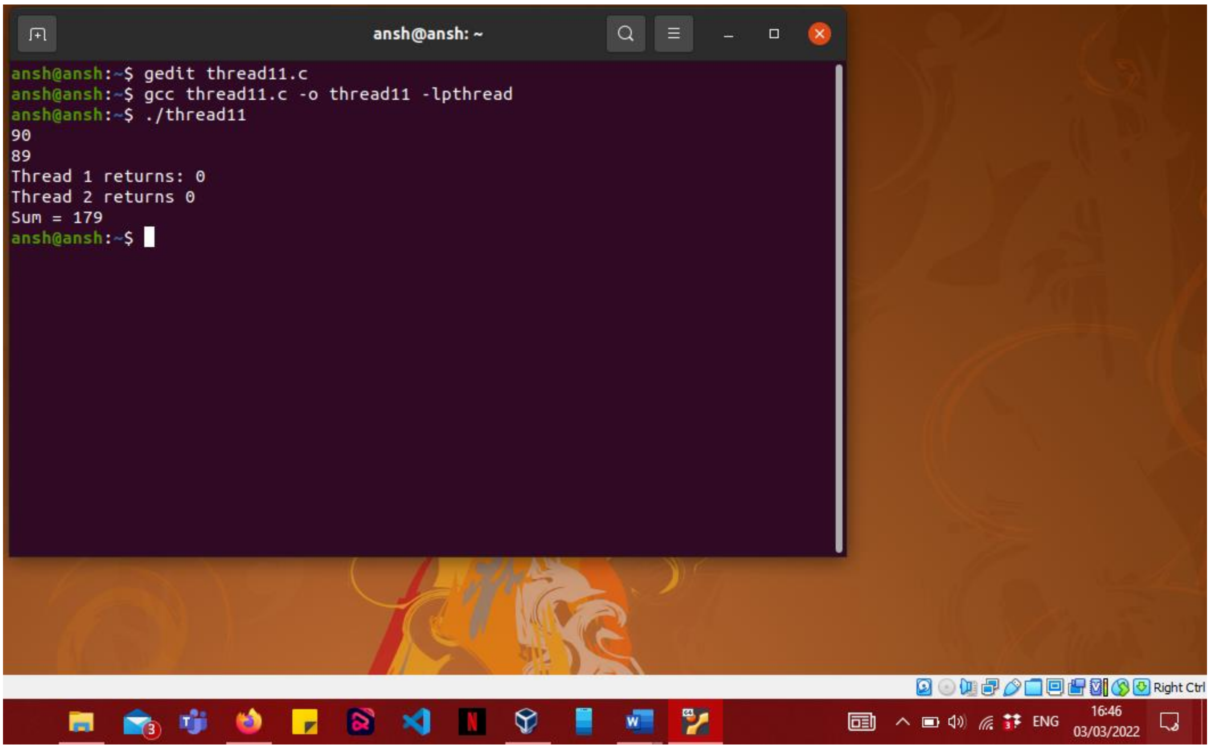
Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<pthread.h>
void *print_message_function();
int sum=0;
int main()
{
pthread_t thread1,thread2;
int iret1, iret2;
//Create independent threads each of which will ececute function
iret1 = pthread_create(&thread1,NULL,print_message_function,NULL);
iret2 = pthread_create(&thread1,NULL,print_message_function,NULL);
pthread_join(thread1,NULL);
pthread_join(thread2,NULL);
printf("Thread 1 returns: %d\n",iret1);
printf("Thread 2 returns %d\n",iret2);
printf("Sum = %d\n",sum);
exit(0);
}
void *print_message_function()
{
int n;
scanf("%d",&n);
sum+=n;
pthread_exit(0);
}
```



```c
#include<stdio.h>
#include<stdlib.h>
#include<pthread.h>
void *print_message_function();
int sum=0;
int main()
{
pthread_t thread1,thread2;
int iret1, iret2;
//Create independent threads each of which will ececute function
iret1 = pthread_create(&thread1,NULL,print_message_function,NULL);
iret2 = pthread_create(&thread1,NULL,print_message_function,NULL);
pthread_join(thread1,NULL);
pthread_join(thread2,NULL);
printf("Thread 1 returns: %d\n",iret1);
printf("Thread 2 returns %d\n",iret2);
printf("Sum = %d\n",sum);
exit(0);
}
void *print_message_function()
{
int n;
scanf("%d",&n);
sum+=n;
pthread_exit(0);
}
```

OUTPUT:



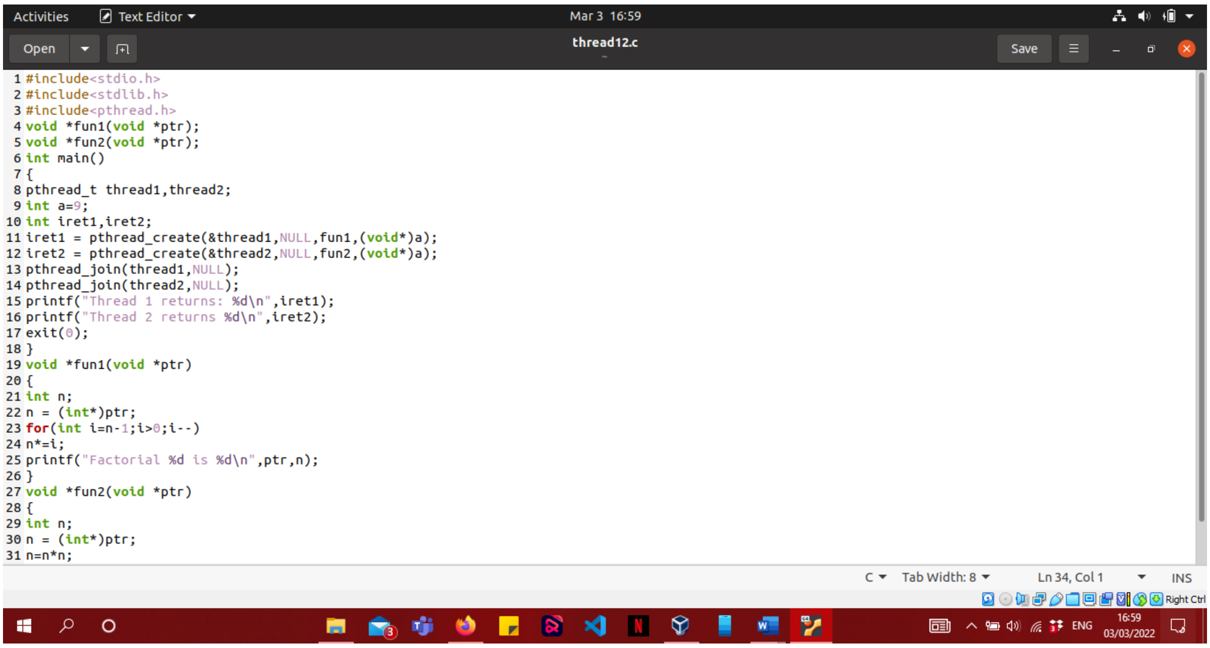
```
ansh@ansh:~$ gedit thread11.c
ansh@ansh:~$ gcc thread11.c -o thread11 -lpthread
ansh@ansh:~$ ./thread11
90
89
Thread 1 returns: 0
Thread 2 returns 0
Sum = 179
ansh@ansh:~$ 
```

Q.2.

Create two threads thread1 and thread2 and call functions fun1 and fun2 respectively. Compute and print factorial of a number in fun1, square of a number in fun2.

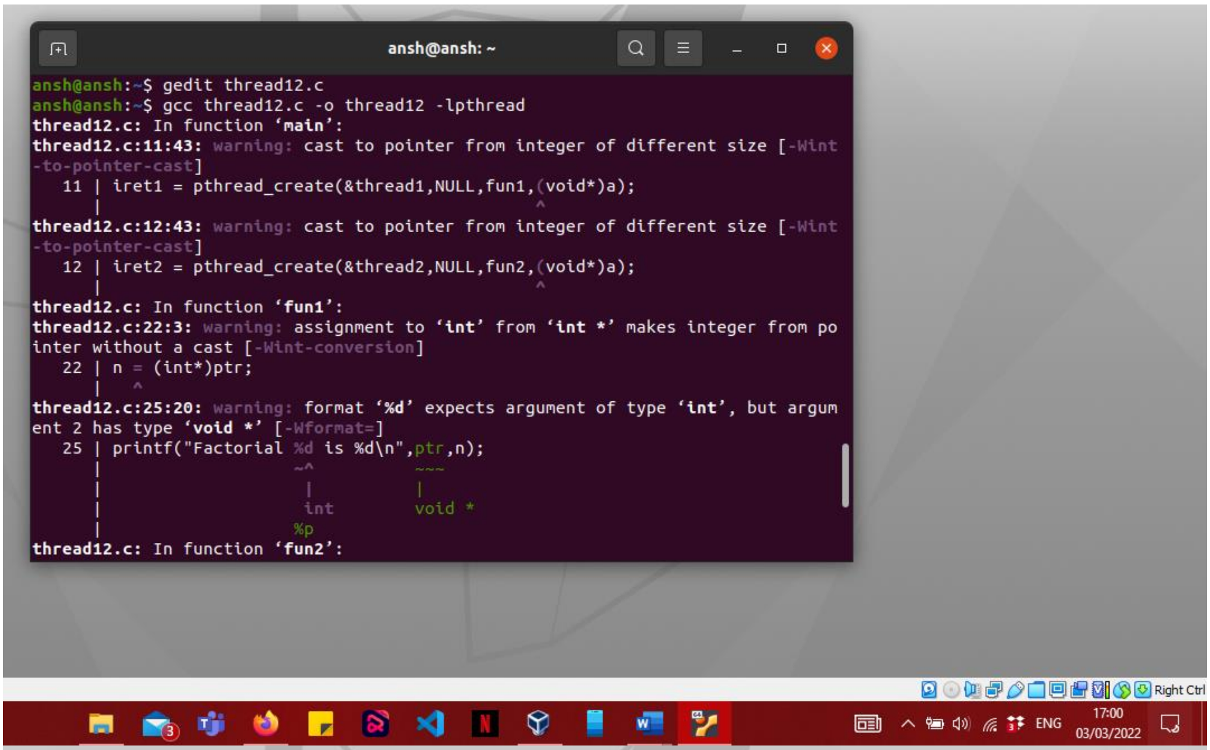
Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<pthread.h>
void *fun1(void *ptr);
void *fun2(void *ptr);
int main()
{
pthread_t thread1,thread2;
int a=9;
int iret1,iret2;
iret1 = pthread_create(&thread1,NULL,fun1,(void*)a);
iret2 = pthread_create(&thread2,NULL,fun2,(void*)a);
pthread_join(thread1,NULL);
pthread_join(thread2,NULL);
printf("Thread 1 returns: %d\n",iret1);
printf("Thread 2 returns %d\n",iret2);
exit(0);
}
void *fun1(void *ptr)
{
int n;
n = (int*)ptr;
for(int i=n-1;i>0;i--)
n*=i;
printf("Factorial %d is %d\n",ptr,n);
}
void *fun2(void *ptr)
{
int n;
n = (int*)ptr;
n=n*n;
printf("Square of %d is %d\n",ptr,n);
}
```

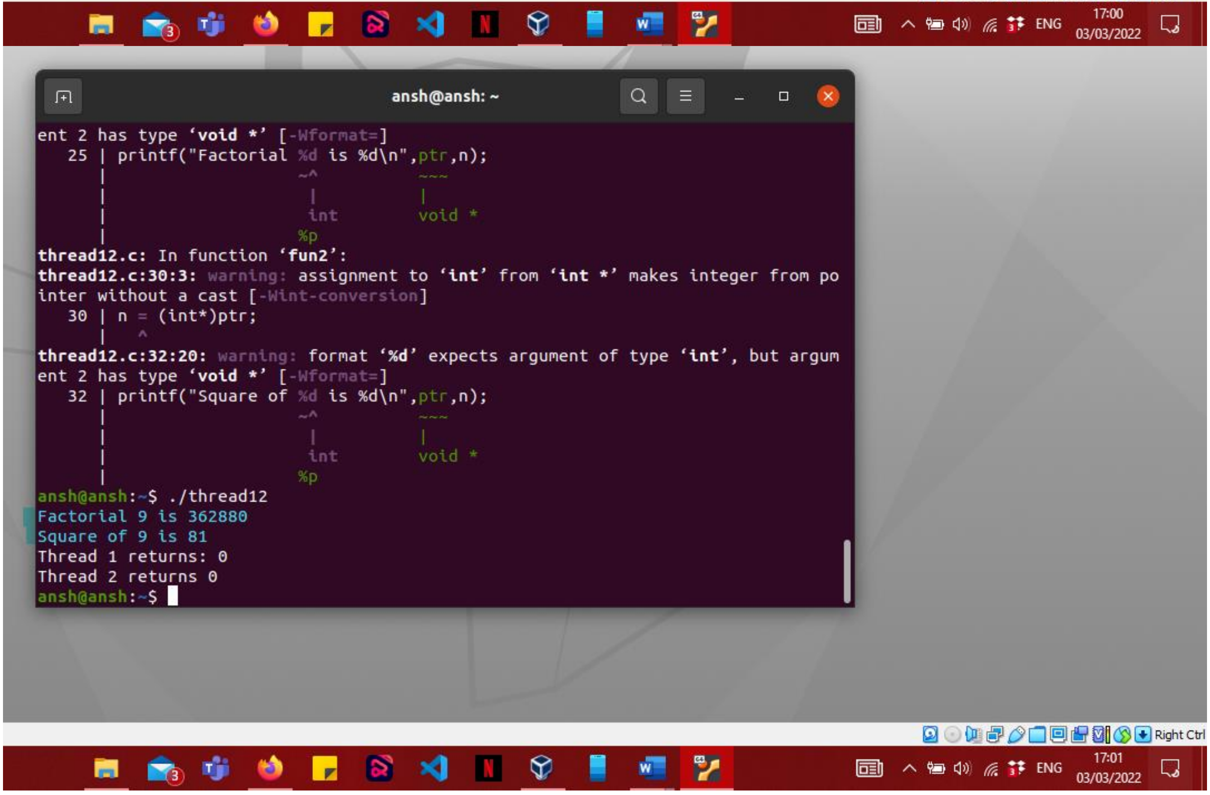


```
#include<stdio.h>
#include<stdlib.h>
#include<pthread.h>
void *fun1(void *ptr);
void *fun2(void *ptr);
int main()
{
pthread_t thread1,thread2;
int a=9;
int iret1,iret2;
iret1 = pthread_create(&thread1,NULL,fun1,(void*)a);
iret2 = pthread_create(&thread2,NULL,fun2,(void*)a);
pthread_join(thread1,NULL);
pthread_join(thread2,NULL);
printf("Thread 1 returns: %d\n",iret1);
printf("Thread 2 returns %d\n",iret2);
exit(0);
}
void *fun1(void *ptr)
{
int n;
n = (int*)ptr;
for(int i=n-1;i>0;i--)
n*=i;
printf("Factorial %d is %d\n",ptr,n);
}
void *fun2(void *ptr)
{
int n;
n = (int*)ptr;
n=n*n;
```

OUTPUT:



```
ansh@ansh:~$ gedit thread12.c
ansh@ansh:~$ gcc thread12.c -o thread12 -lpthread
thread12.c: In function ‘main’:
thread12.c:11:43: warning: cast to pointer from integer of different size [-Wint
-to-pointer-cast]
   11 | iret1 = pthread_create(&thread1,NULL,fun1,(void*)a);
      |                                           ^
thread12.c:12:43: warning: cast to pointer from integer of different size [-Wint
-to-pointer-cast]
   12 | iret2 = pthread_create(&thread2,NULL,fun2,(void*)a);
      |                                           ^
thread12.c: In function ‘fun1’:
thread12.c:22:3: warning: assignment to ‘int’ from ‘int *’ makes integer from po
inter without a cast [-Wint-conversion]
   22 | n = (int*)ptr;
      |   ^
thread12.c:25:20: warning: format ‘%d’ expects argument of type ‘int’, but argum
ent 2 has type ‘void *’ [-Wformat=]
   25 | printf("Factorial %d is %d\n",ptr,n);
      |                   ~^           ~~~
      |                    |           |
      |                    int         void *
      |                   %p
thread12.c: In function ‘fun2’:
```



```
ent 2 has type ‘void *’ [-Wformat=]
   25 | printf("Factorial %d is %d\n",ptr,n);
      |                   ~^           ~~~
      |                    |           |
      |                    int         void *
      |                   %p
thread12.c: In function ‘fun2’:
thread12.c:30:3: warning: assignment to ‘int’ from ‘int *’ makes integer from po
inter without a cast [-Wint-conversion]
   30 | n = (int*)ptr;
      |   ^
thread12.c:32:20: warning: format ‘%d’ expects argument of type ‘int’, but argum
ent 2 has type ‘void *’ [-Wformat=]
   32 | printf("Square of %d is %d\n",ptr,n);
      |                   ~^           ~~~
      |                    |           |
      |                    int         void *
      |                   %p
ansh@ansh:~$ ./thread12
Factorial 9 is 362880
Square of 9 is 81
Thread 1 returns: 0
Thread 2 returns 0
ansh@ansh:~$ 
```