

# Operating System

## Lab Report

### Lab 10



Group Members Name & Reg #:	<b><u>Muhammad Haris Irfan</u></b> <b>(FA18-BCE-090)</b>
Class	Operating System ( <b>BCE-5B</b> )
Instructor's Name	Sir Wasif

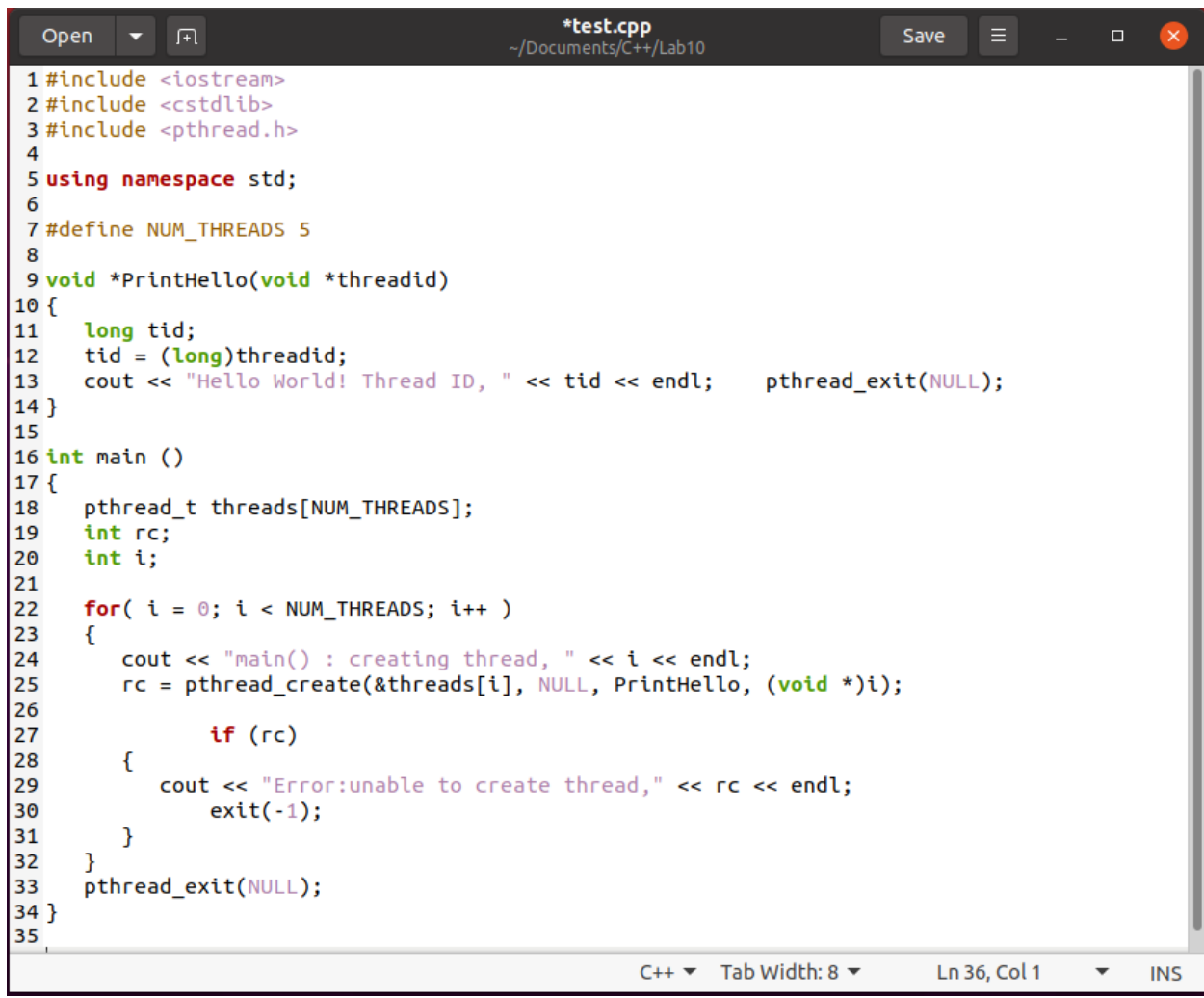
# In Lab Tasks

## Question 1: Activity 1:

Write a program in C++ to create and terminate threads.

### Solution:

➤ Code is given below:

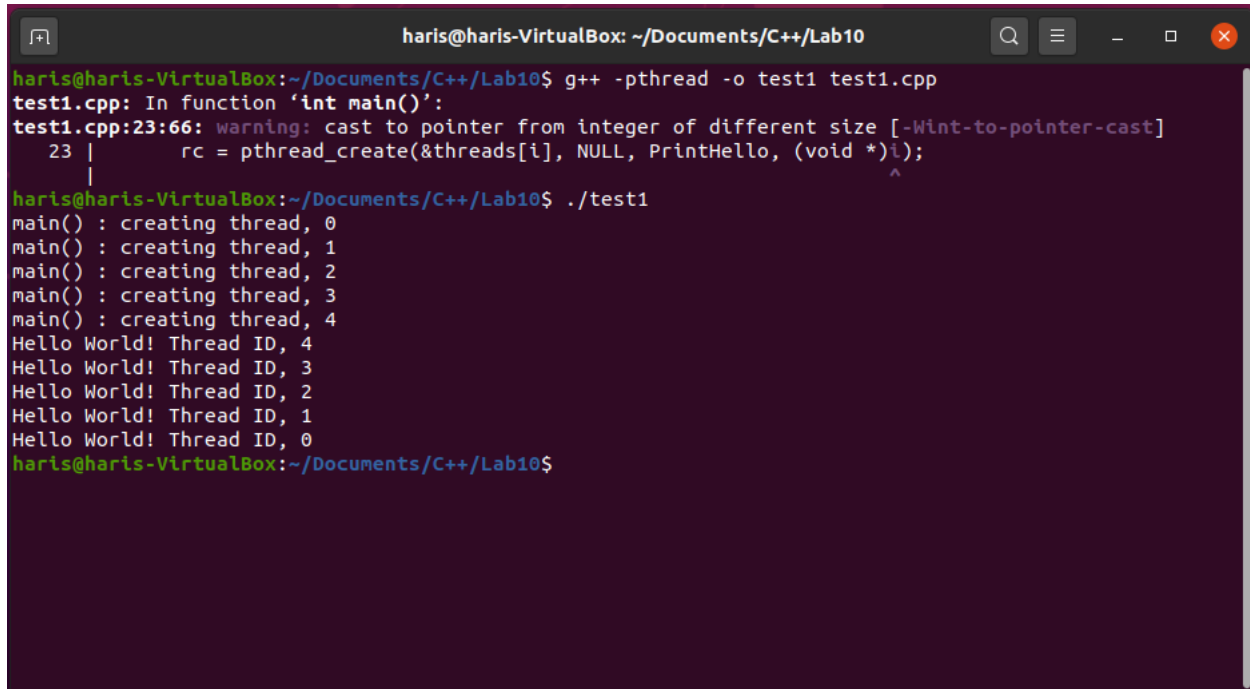


```
*test.cpp
~/Documents/C++/Lab10
Save

1 #include <iostream>
2 #include <cstdlib>
3 #include <pthread.h>
4
5 using namespace std;
6
7 #define NUM_THREADS 5
8
9 void *PrintHello(void *threadid)
10 {
11     long tid;
12     tid = (long)threadid;
13     cout << "Hello World! Thread ID, " << tid << endl;    pthread_exit(NULL);
14 }
15
16 int main ()
17 {
18     pthread_t threads[NUM_THREADS];
19     int rc;
20     int i;
21
22     for( i = 0; i < NUM_THREADS; i++ )
23     {
24         cout << "main() : creating thread, " << i << endl;
25         rc = pthread_create(&threads[i], NULL, PrintHello, (void *)i);
26
27         if (rc)
28         {
29             cout << "Error:unable to create thread," << rc << endl;
30             exit(-1);
31         }
32     }
33     pthread_exit(NULL);
34 }
35
```

C++ Tab Width: 8 Ln 36, Col 1 INS

The Terminal Output is given below,

A terminal window titled 'haris@haris-VirtualBox: ~/Documents/C++/Lab10'. The prompt is 'haris@haris-VirtualBox:~/Documents/C++/Lab10\$'. The user enters 'g++ -pthread -o test1 test1.cpp'. The output shows a warning: 'test1.cpp: In function 'int main()': test1.cpp:23:66: warning: cast to pointer from integer of different size [-Wint-to-pointer-cast]'. The code snippet shows 'rc = pthread\_create(&threads[i], NULL, PrintHello, (void \*)i);' with a caret under the 'i'. The user then enters './test1'. The output shows 'main() : creating thread, 0' through 'main() : creating thread, 4', followed by 'Hello World! Thread ID, 4' through 'Hello World! Thread ID, 0'. The prompt returns to 'haris@haris-VirtualBox:~/Documents/C++/Lab10\$'.

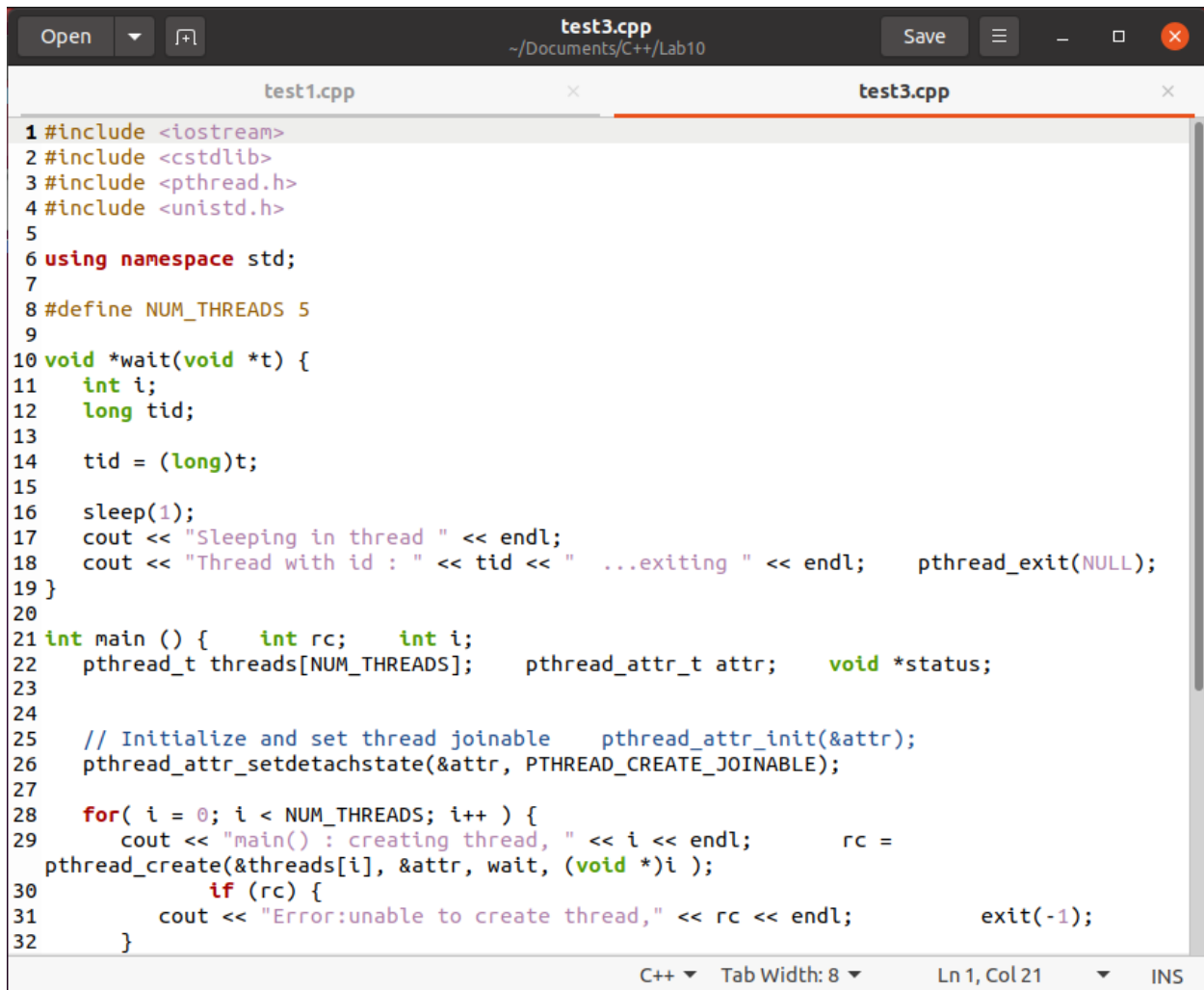
```
haris@haris-VirtualBox: ~/Documents/C++/Lab10
haris@haris-VirtualBox:~/Documents/C++/Lab10$ g++ -pthread -o test1 test1.cpp
test1.cpp: In function 'int main()':
test1.cpp:23:66: warning: cast to pointer from integer of different size [-Wint-to-pointer-cast]
   23 |         rc = pthread_create(&threads[i], NULL, PrintHello, (void *)i);
       |                                                                    ^
haris@haris-VirtualBox:~/Documents/C++/Lab10$ ./test1
main() : creating thread, 0
main() : creating thread, 1
main() : creating thread, 2
main() : creating thread, 3
main() : creating thread, 4
Hello World! Thread ID, 4
Hello World! Thread ID, 3
Hello World! Thread ID, 2
Hello World! Thread ID, 1
Hello World! Thread ID, 0
haris@haris-VirtualBox:~/Documents/C++/Lab10$
```

## Question 2: Activity 2:

### Passing Arguments to Threads

### Solution:

➤ Code is given below:

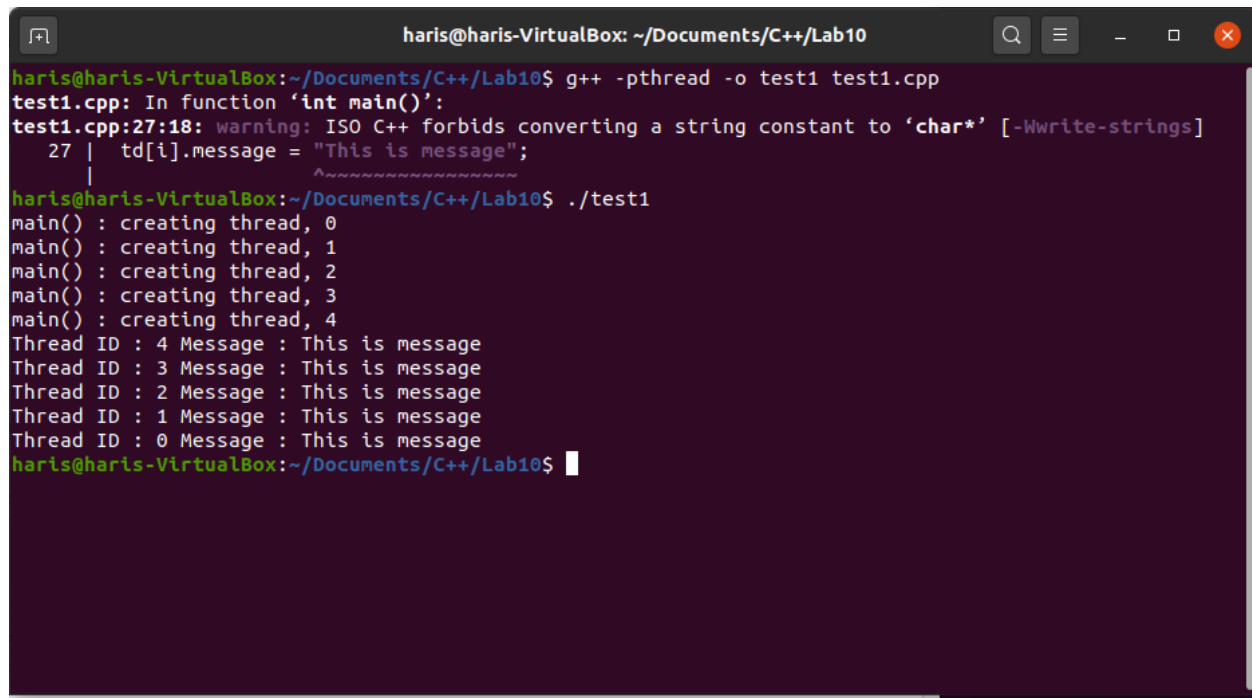


The screenshot shows a code editor window with two tabs: 'test1.cpp' and 'test3.cpp'. The 'test3.cpp' tab is active and displays the following C++ code:

```
1 #include <iostream>
2 #include <cstdlib>
3 #include <pthread.h>
4 #include <unistd.h>
5
6 using namespace std;
7
8 #define NUM_THREADS 5
9
10 void *wait(void *t) {
11     int i;
12     long tid;
13
14     tid = (long)t;
15
16     sleep(1);
17     cout << "Sleeping in thread " << endl;
18     cout << "Thread with id : " << tid << " ...exiting " << endl;    pthread_exit(NULL);
19 }
20
21 int main () {    int rc;    int i;
22     pthread_t threads[NUM_THREADS];    pthread_attr_t attr;    void *status;
23
24
25     // Initialize and set thread joinable    pthread_attr_init(&attr);
26     pthread_attr_setdetachstate(&attr, PTHREAD_CREATE_JOINABLE);
27
28     for( i = 0; i < NUM_THREADS; i++ ) {
29         cout << "main() : creating thread, " << i << endl;        rc =
pthread_create(&threads[i], &attr, wait, (void *)i );
30         if (rc) {
31             cout << "Error:unable to create thread," << rc << endl;        exit(-1);
32     }
```

The editor's status bar at the bottom indicates 'C++', 'Tab Width: 8', 'Ln 1, Col 21', and 'INS'.

The Terminal Output is given below,

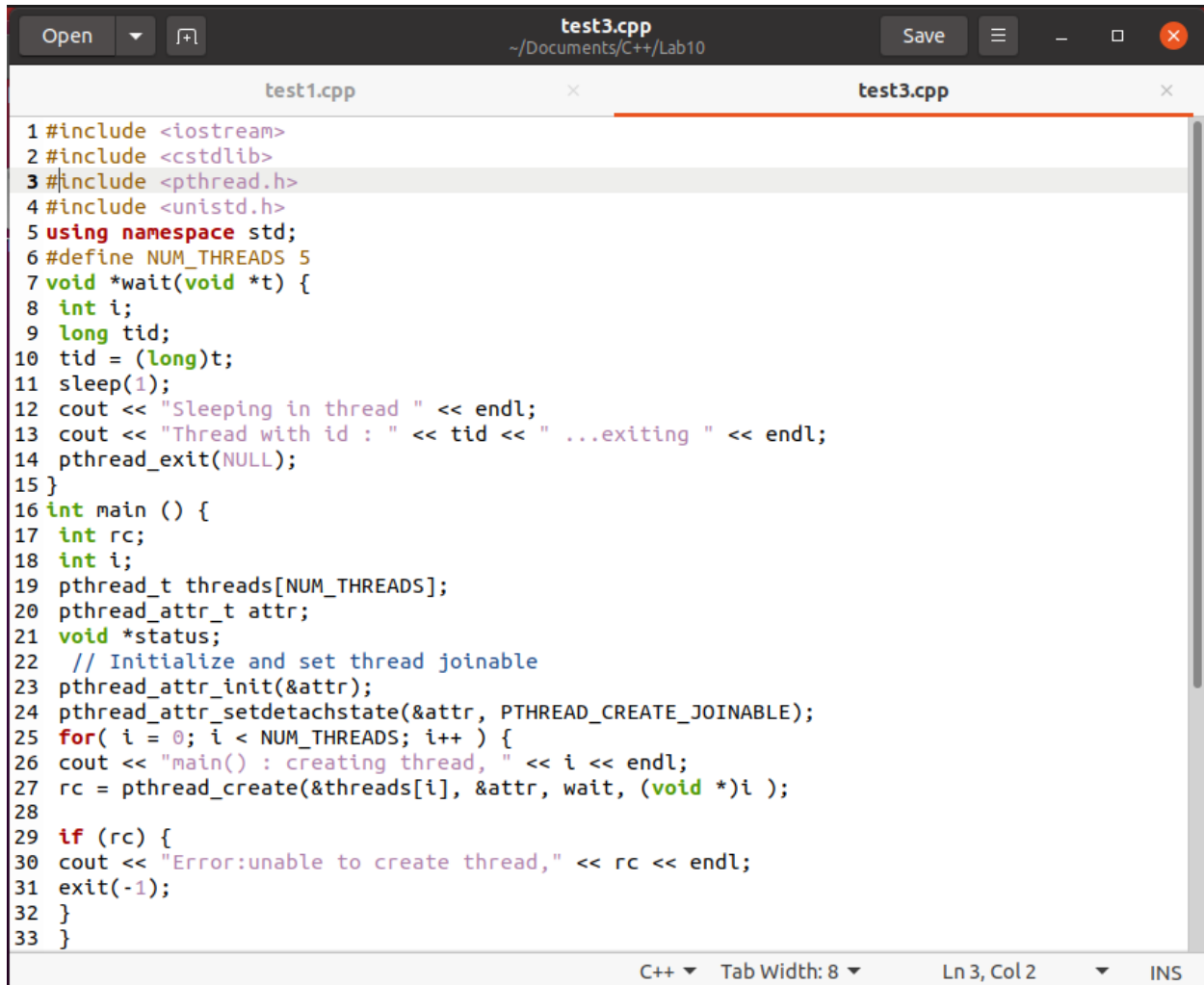
A terminal window titled 'haris@haris-VirtualBox: ~/Documents/C++/Lab10'. The prompt is 'haris@haris-VirtualBox:~/Documents/C++/Lab10\$'. The first command is 'g++ -pthread -o test1 test1.cpp'. The output shows a warning: 'test1.cpp: In function 'int main()': test1.cpp:27:18: warning: ISO C++ forbids converting a string constant to 'char\*' [-Wwrite-strings]'. The code snippet shows 'td[i].message = "This is message";'. The second command is './test1'. The output shows 'main() : creating thread, 0' through 'main() : creating thread, 4', followed by 'Thread ID : 4 Message : This is message' through 'Thread ID : 0 Message : This is message'. The prompt returns to 'haris@haris-VirtualBox:~/Documents/C++/Lab10\$'.

## Question 3: Activity 3:

Joining and Detaching Threads

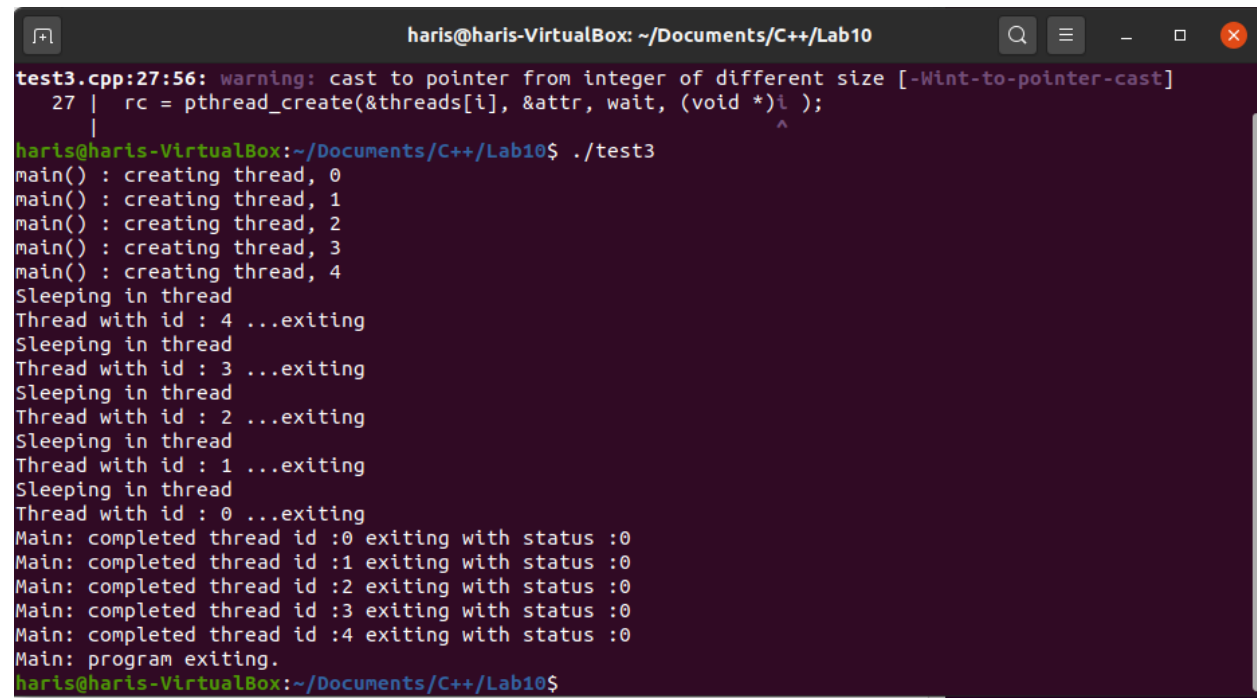
### Solution:

➤ Code is given below:



```
1 #include <iostream>
2 #include <cstdlib>
3 #include <pthread.h>
4 #include <unistd.h>
5 using namespace std;
6 #define NUM_THREADS 5
7 void *wait(void *t) {
8     int i;
9     long tid;
10    tid = (long)t;
11    sleep(1);
12    cout << "Sleeping in thread " << endl;
13    cout << "Thread with id : " << tid << " ...exiting " << endl;
14    pthread_exit(NULL);
15 }
16 int main () {
17     int rc;
18     int i;
19     pthread_t threads[NUM_THREADS];
20     pthread_attr_t attr;
21     void *status;
22     // Initialize and set thread joinable
23     pthread_attr_init(&attr);
24     pthread_attr_setdetachstate(&attr, PTHREAD_CREATE_JOINABLE);
25     for( i = 0; i < NUM_THREADS; i++ ) {
26         cout << "main() : creating thread, " << i << endl;
27         rc = pthread_create(&threads[i], &attr, wait, (void *)i );
28     }
29     if (rc) {
30         cout << "Error:unable to create thread," << rc << endl;
31         exit(-1);
32     }
33 }
```

The Terminal Output is given below,

A terminal window titled 'haris@haris-VirtualBox: ~/Documents/C++/Lab10'. It shows the compilation of 'test3.cpp' with a warning about a cast from integer to pointer. Then, the program is executed, showing the creation of 5 threads (IDs 0-4), each sleeping and then exiting. The main function waits for all threads to complete before exiting.

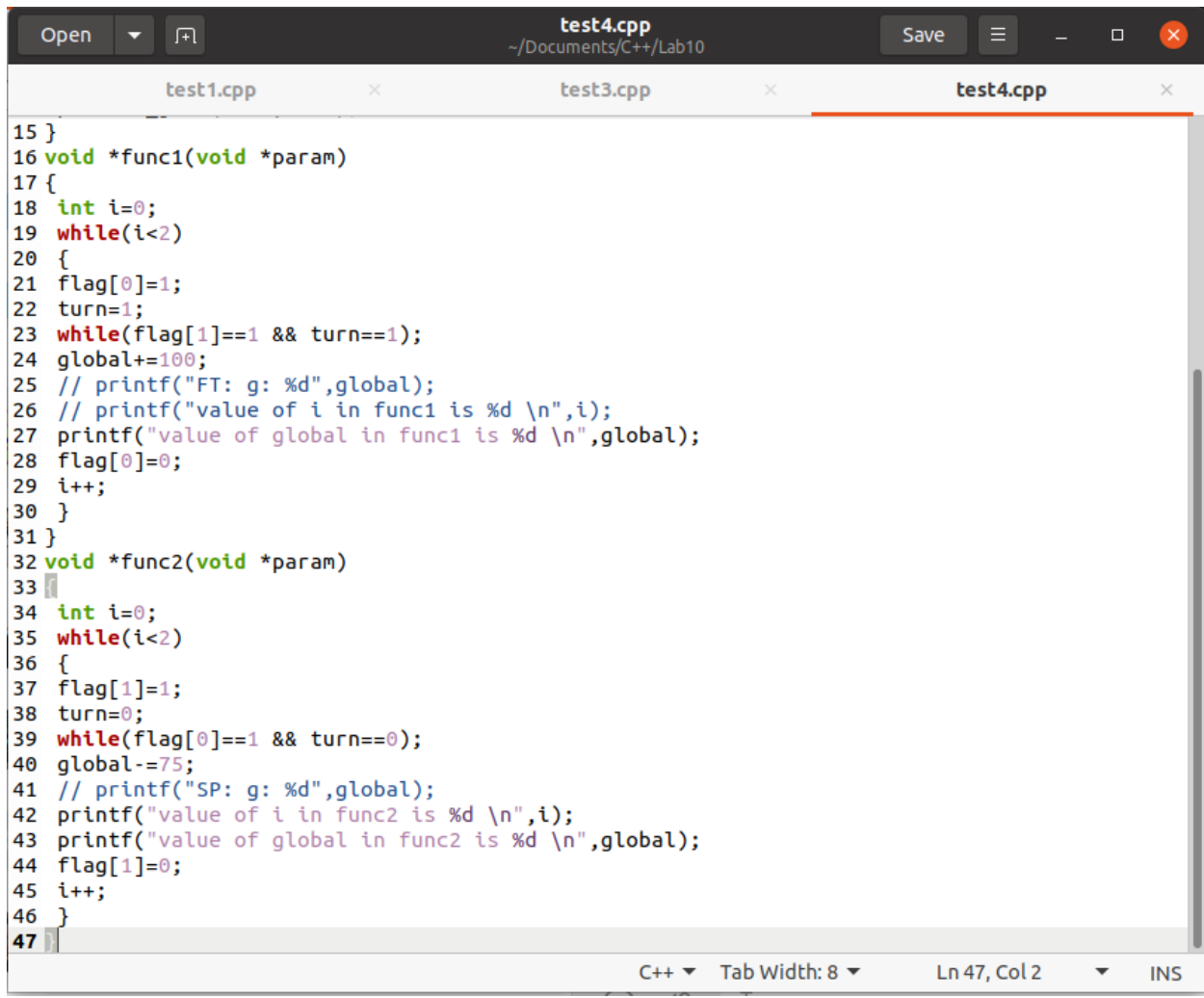
```
haris@haris-VirtualBox: ~/Documents/C++/Lab10
test3.cpp:27:56: warning: cast to pointer from integer of different size [-Wint-to-pointer-cast]
 27 | rc = pthread_create(&threads[i], &attr, wait, (void *)i );
    |                                     ^
haris@haris-VirtualBox:~/Documents/C++/Lab10$ ./test3
main() : creating thread, 0
main() : creating thread, 1
main() : creating thread, 2
main() : creating thread, 3
main() : creating thread, 4
Sleeping in thread
Thread with id : 4 ...exiting
Sleeping in thread
Thread with id : 3 ...exiting
Sleeping in thread
Thread with id : 2 ...exiting
Sleeping in thread
Thread with id : 1 ...exiting
Sleeping in thread
Thread with id : 0 ...exiting
Main: completed thread id :0 exiting with status :0
Main: completed thread id :1 exiting with status :0
Main: completed thread id :2 exiting with status :0
Main: completed thread id :3 exiting with status :0
Main: completed thread id :4 exiting with status :0
Main: program exiting.
haris@haris-VirtualBox:~/Documents/C++/Lab10$
```

## Question 4: Activity part 1:

Peterson solution for 2 processes critical section problem

### Solution:

➤ Code is given below:



```
test4.cpp
~/Documents/C++/Lab10
Save

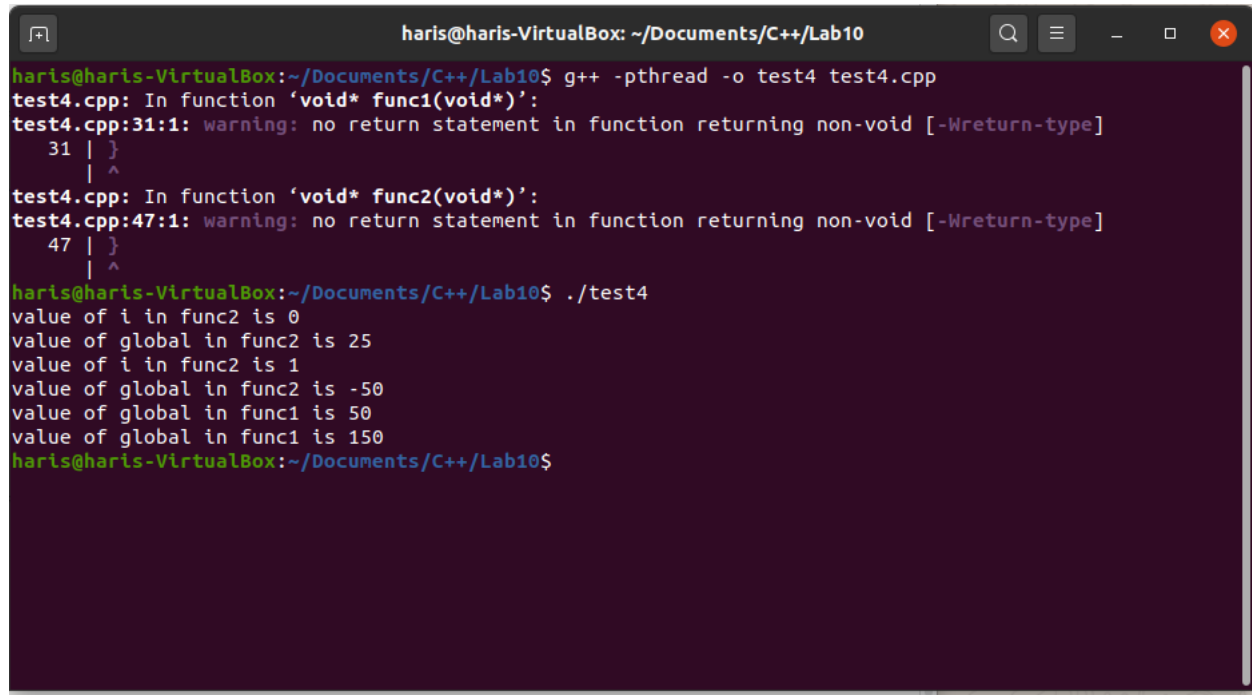
test1.cpp  test3.cpp  test4.cpp

15 }
16 void *func1(void *param)
17 {
18     int i=0;
19     while(i<2)
20     {
21         flag[0]=1;
22         turn=1;
23         while(flag[1]==1 && turn==1);
24         global+=100;
25         // printf("FT: g: %d",global);
26         // printf("value of i in func1 is %d \n",i);
27         printf("value of global in func1 is %d \n",global);
28         flag[0]=0;
29         i++;
30     }
31 }
32 void *func2(void *param)
33 {
34     int i=0;
35     while(i<2)
36     {
37         flag[1]=1;
38         turn=0;
39         while(flag[0]==1 && turn==0);
40         global-=75;
41         // printf("SP: g: %d",global);
42         printf("value of i in func2 is %d \n",i);
43         printf("value of global in func2 is %d \n",global);
44         flag[1]=0;
45         i++;
46     }
47 }
```

C++ Tab Width: 8 Ln 47, Col 2 INS



The Terminal Output is given below,

A terminal window titled 'haris@haris-VirtualBox: ~/Documents/C++/Lab10' with standard window controls. The terminal shows the compilation of 'test4.cpp' using 'g++ -pthread -o test4 test4.cpp'. It displays two warnings: 'warning: no return statement in function returning non-void [-Wreturn-type]' for functions 'func1' and 'func2'. The execution command './test4' is run, producing output for variables 'i' and 'global' in both functions. The output shows that 'i' starts at 0 and increments to 1, while 'global' starts at 25 and increments to 150 across the two function calls.

```
haris@haris-VirtualBox: ~/Documents/C++/Lab10
haris@haris-VirtualBox:~/Documents/C++/Lab10$ g++ -pthread -o test4 test4.cpp
test4.cpp: In function 'void* func1(void*)':
test4.cpp:31:1: warning: no return statement in function returning non-void [-Wreturn-type]
   31 | }
      | ^
test4.cpp: In function 'void* func2(void*)':
test4.cpp:47:1: warning: no return statement in function returning non-void [-Wreturn-type]
   47 | }
      | ^
haris@haris-VirtualBox:~/Documents/C++/Lab10$ ./test4
value of i in func2 is 0
value of global in func2 is 25
value of i in func2 is 1
value of global in func2 is -50
value of global in func1 is 50
value of global in func1 is 150
haris@haris-VirtualBox:~/Documents/C++/Lab10$
```

## Question 4: Activity part 2:

### Modified Code

### Solution:

➤ Code is given below:

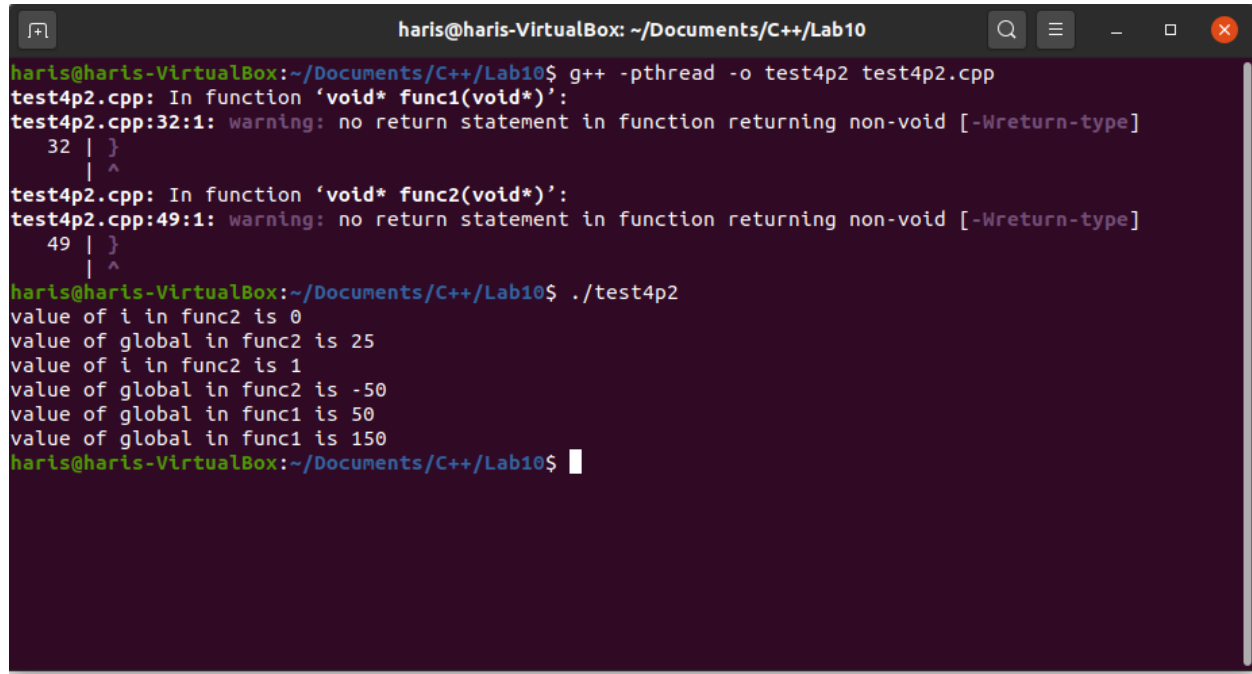


```
test4p2.cpp
~/Documents/C++/Lab10

1 #include<pthread.h>
2 #include<stdio.h>
3 void *func1(void *);
4 void *func2(void *);
5 int flag[2];
6 int turn=0;
7 int global=100;
8 int main()
9 {
10 pthread_t tid1,tid2;
11 pthread_create(&tid1,NULL,func1,NULL);
12 pthread_create(&tid2,NULL,func2,NULL);
13 pthread_join(tid1,NULL);
14 pthread_join(tid2,NULL);
15 }
16 void *func1(void *param)
17 {
18 int i=0;
19 while(i<2)
20 {
21 flag[0]=1;
22 turn=1;
23 __sync_synchronize();
24 while(flag[1]==1 && turn==1) sched_yield();
25 global+=100;
26 // printf("FT: g: %d",global);
27 // printf("value of i in func1 is %d \n",i);
28 printf("value of global in func1 is %d \n",global);
29 flag[0]=0;
30 i++;
31 }
32 }
33 void *func2(void *param)
34 {
35 int i=0;
36 while(i<2)
```

C++ Tab Width: 8 Ln 17, Col 2 INS

The Terminal Output is given below,

A terminal window titled 'haris@haris-VirtualBox: ~/Documents/C++/Lab10'. The prompt is 'haris@haris-VirtualBox:~/Documents/C++/Lab10\$'. The first command is 'g++ -pthread -o test4p2 test4p2.cpp'. The output shows two warnings: 'test4p2.cpp: In function 'void\* func1(void\*)': test4p2.cpp:32:1: warning: no return statement in function returning non-void [-Wreturn-type]' and 'test4p2.cpp: In function 'void\* func2(void\*)': test4p2.cpp:49:1: warning: no return statement in function returning non-void [-Wreturn-type]'. The second command is './test4p2'. The output shows: 'value of i in func2 is 0', 'value of global in func2 is 25', 'value of i in func2 is 1', 'value of global in func2 is -50', 'value of global in func1 is 50', 'value of global in func1 is 150'. The prompt is 'haris@haris-VirtualBox:~/Documents/C++/Lab10\$' with a cursor.

```
haris@haris-VirtualBox:~/Documents/C++/Lab10$ g++ -pthread -o test4p2 test4p2.cpp
test4p2.cpp: In function 'void* func1(void*)':
test4p2.cpp:32:1: warning: no return statement in function returning non-void [-Wreturn-type]
   32 | }
      | ^
test4p2.cpp: In function 'void* func2(void*)':
test4p2.cpp:49:1: warning: no return statement in function returning non-void [-Wreturn-type]
   49 | }
      | ^
haris@haris-VirtualBox:~/Documents/C++/Lab10$ ./test4p2
value of i in func2 is 0
value of global in func2 is 25
value of i in func2 is 1
value of global in func2 is -50
value of global in func1 is 50
value of global in func1 is 150
haris@haris-VirtualBox:~/Documents/C++/Lab10$
```

\_\_\_\_\_THE END\_\_\_\_\_

=====