LAB 8
Roll Num: 23k-0062
Name: Syed Fahad Faheem Shah

Observations and Discussion

```
a) pthread_create() function  pass argument num in runner function
```

```
b) in this lab we see how default attributes are initialized
```

```
c)
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>

void *thread_function(void *args) {
   printf("Thread Executing .... \n");
   pthread_exit((void *)42);
}

int main(int argc, char *argv[]) {
   pthread_t threadID;
   void * exit_status;

   pthread_create(&threadID, NULL, thread_function,NULL);
   pthread_join(threadID, &exit_status);
   printf("Thread exited with status: %ld\n", (long)exit_status);
   return 0;
}
```

Output

```
Executing ....
exited with status: 42
t@student-OptiPlex-7090:~/Lab 8$
```

Code Workout 2

a)

```
fahad@Fahad-Shah:~/Desktop/Threads/Lab_Tasks$ gcc inlab2.c -o inlab2.o -pthread
fahad@Fahad-Shah:~/Desktop/Threads/Lab_Tasks$ ./inlab2.o
main: begin (counter = 0)
A: begin
B: begin
B: done. Counter = 12156563
A: done. Counter = 12968112
main: done with both (counter = 12968112)
fahad@Fahad-Shah:~/Desktop/Threads/Lab_Tasks$
```

b) Because counter is a global variable when thread 1 is done with their work it will store the value of counter, so when thread 2 starts their work it will add in counter value not from zero but from value where thread 1 ends it.

C) After uncomment line 9 output from both A and B are same B/C counter is set to 0 in function.

```
fahad@Fahad-Shah:~/Desktop/Threads/Lab_Tasks$ gcc inlab2.c -o inlab2.o -pthread
fahad@Fahad-Shah:~/Desktop/Threads/Lab_Tasks$ ./inlab2.o
main: begin (counter = 0)
A: begin
B: begin
B: done. Counter = 10000000
A: done. Counter = 10000000
main: done with both (counter = 0)
fahad@Fahad-Shah:~/Desktop/Threads/Lab_Tasks$ s
```