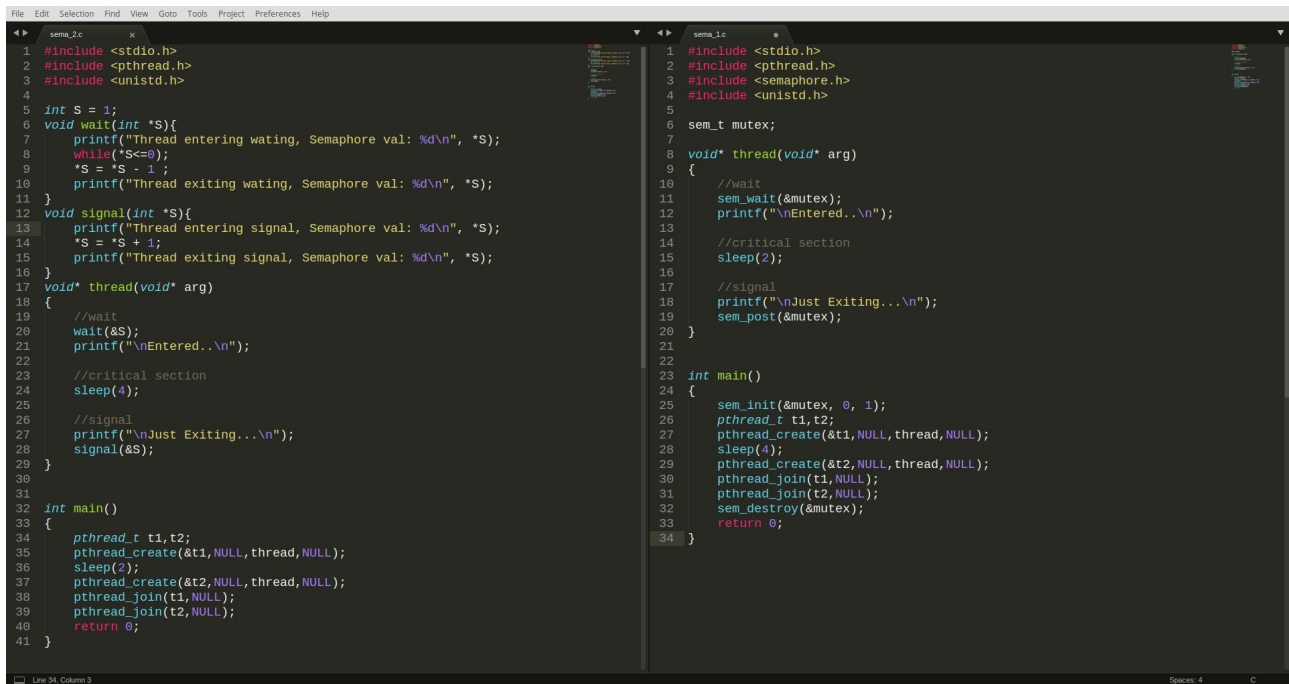


# SEMAPHORE

JATIN KARTHIK TRIPATHY  
17BCE7106



```
File Edit Selection Find View Goto Tools Project Preferences Help
sema_2.c
1 #include <stdio.h>
2 #include <pthread.h>
3 #include <unistd.h>
4
5 int S = 1;
6 void wait(int *S){
7     printf("Thread entering wating, Semaphore val: %d\n", *S);
8     while(*S<=0);
9     *S = *S - 1;
10    printf("Thread exiting wating, Semaphore val: %d\n", *S);
11 }
12 void signal(int *S){
13     printf("Thread entering signal, Semaphore val: %d\n", *S);
14     *S = *S + 1;
15     printf("Thread exiting signal, Semaphore val: %d\n", *S);
16 }
17 void* thread(void* arg)
18 {
19     //wait
20     wait(&S);
21     printf("\nEntered..\n");
22
23     //critical section
24     sleep(4);
25
26     //signal
27     printf("\nJust Exiting...\n");
28     signal(&S);
29 }
30
31 int main()
32 {
33     pthread_t t1,t2;
34     pthread_create(&t1,NULL,thread,NULL);
35     sleep(2);
36     pthread_create(&t2,NULL,thread,NULL);
37     pthread_join(t1,NULL);
38     pthread_join(t2,NULL);
39     return 0;
40 }
41
sema_1.c
1 #include <stdio.h>
2 #include <pthread.h>
3 #include <semaphore.h>
4 #include <unistd.h>
5
6 sem_t mutex;
7
8 void* thread(void* arg)
9 {
10    //wait
11    sem_wait(&mutex);
12    printf("\nEntered..\n");
13
14    //critical section
15    sleep(2);
16
17    //signal
18    printf("\nJust Exiting...\n");
19    sem_post(&mutex);
20 }
21
22 int main()
23 {
24     sem_init(&mutex, 0, 1);
25     pthread_t t1,t2;
26     pthread_create(&t1,NULL,thread,NULL);
27     sleep(4);
28     pthread_create(&t2,NULL,thread,NULL);
29     pthread_join(t1,NULL);
30     pthread_join(t2,NULL);
31     sem_destroy(&mutex);
32     return 0;
33 }
34
```

```
semaphore: ./sema_2
+ .bin/caa x semaphore: ./sema_2
jkt@jkt-pc:/media/jkt/New Volume/OS_LAB/17BCE7106/semaphore$ gcc -o sema_1 sema_1.c -lpthread
jkt@jkt-pc:/media/jkt/New Volume/OS_LAB/17BCE7106/semaphore$ ./sema_1

Entered..

Just Exiting...

Entered..

Just Exiting...
jkt@jkt-pc:/media/jkt/New Volume/OS_LAB/17BCE7106/semaphore$ gcc -o sema_2 sema_2.c -lpthread
jkt@jkt-pc:/media/jkt/New Volume/OS_LAB/17BCE7106/semaphore$ ./sema_2
Thread entering wating, Semaphore val: 1
Thread exiting wating, Semaphore val: 0

Entered..
Thread entering wating, Semaphore val: 0

Just Exiting...
Thread entering signal, Semaphore val: 0
Thread exiting signal, Semaphore val: 1
Thread exiting wating, Semaphore val: 0

Entered..

Just Exiting...
Thread entering signal, Semaphore val: 0
Thread exiting signal, Semaphore val: 1
jkt@jkt-pc:/media/jkt/New Volume/OS_LAB/17BCE7106/semaphore$
```