



main.c



Share

Run

Output

Clear

```
1 #include <stdio.h>
2 #include <pthread.h>
3 #include <semaphore.h>
4 #define MAX 5
5 sem_t mutex;
6 int counter = 0;
7 void* process(void* arg) {
8     sem_wait(&mutex);
9     counter++;
10    printf("Process %d: Counter = %d\n", *(int*)arg, counter);
11    sem_post(&mutex);
12    return NULL;
13 }
14 int main() {
15    pthread_t threads[MAX];
16    int process_ids[MAX];
17    sem_init(&mutex, 0, 1);
18    for (int i = 0; i < MAX; i++) {
19        process_ids[i] = i + 1;
20        pthread_create(&threads[i], NULL, process, (void
21        *)&process_ids[i]);
22    }
```

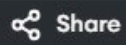
```
Process 3: Counter = 1
Process 2: Counter = 2
Process 4: Counter = 3
Process 1: Counter = 4
Process 5: Counter = 5
```

```
=== Code Execution Successful ===
```





main.c



Run

Output

Clear

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <pthread.h>
4
5 void* threadFunction(void* arg) {
6     printf("Thread is running\n");
7     return NULL;
8 }
9
10 int main() {
11     pthread_t thread;
12     pthread_create(&thread, NULL, threadFunction, NULL);
13     pthread_join(thread, NULL);
14     return 0;
15 }
16
```

Thread is running

=== Code Execution Successful ===

JS

GO

PHP

Light rain
Today



Search



ENG
IN



9:13 AM
12/17/2024