


```

1 #include <stdio.h>
2 #include <conio.h>
3 int main()
4 {
5     int i, NOP, sum=0, count=0, y, quant, wt=0, tat=0, at[10], bt[10], temp[10];
6     float avg_wt, avg_tat;
7     printf("Total number of process in the system: ");
8     scanf("%d", &NOP);
9     y = NOP;
10    for(i=0; i<NOP; i++)
11    {
12        printf("Enter the Arrival and Burst time of the Process[%d]\n", i+1);
13        printf("Arrival time is: ");
14        scanf("%d", &at[i]);
15        printf("Burst time is: ");
16        scanf("%d", &bt[i]);
17        temp[i] = bt[i];
18    }
19    printf("Enter the Time Quantum for the process: ");
20    scanf("%d", &quant);
21    printf("Process No \t\t Burst Time \t\t TAT\t\t Waiting Time ");
22    for(sum=0, i = 0; y!=0; )
23    {
24        if(temp[i] <= quant && temp[i] > 0)
25        {
26            sum = sum + temp[i];
27            temp[i] = 0;
28            count++;
29        }
30        else if(temp[i] > 0)
31        {
32            temp[i] = temp[i] - quant;
33            sum = sum + quant;

```

Resources Compile Log Debug Find Results Console Close

```

- Output Filename: C:\Users\aswin\Documents\ROUNDROBIN.
- Output Size: 323.7841796875 KiB
- Compilation Time: 0.19s

```

piler patl

C:\Users\aswin\Documents\ROUNDROBIN.exe

```

Total number of process in the system: 4
Enter the Arrival and Burst time of the Process[1]
Arrival time is: 0
Burst time is: 15
Enter the Arrival and Burst time of the Process[2]
Arrival time is: 8
Burst time is: 21
Enter the Arrival and Burst time of the Process[3]
Arrival time is: 9
Burst time is: 31
Enter the Arrival and Burst time of the Process[4]
Arrival time is: 18
Burst time is: 22
Enter the Time Quantum for the process: 6

```

| Process No | Burst Time | TAT | Waiting Time |
|---------------|------------|-----|--------------|
| Process No[1] | 15 | 33 | 18 |
| Process No[2] | 21 | 64 | 43 |
| Process No[4] | 22 | 64 | 42 |
| Process No[3] | 31 | 80 | 49 |

Average Turn Around Time: 38.000000

Average Waiting Time: 60.250000

Process exited after 19.25 seconds with return value 0

Press any key to continue . . .

```
ipc.cpp x multithreading.cpp x
1  #include <stdio.h>
2  #include <string.h>
3  #include <pthread.h>
4  int i = 2;
5
6  void* foo(void* p){
7      printf("Value received as argument in starting routine: ");
8      printf("%i\n", * (int*)p);
9      pthread_exit(&i);
10 }
11
12 int main(void){
13     pthread_t id;
14     int j = 1;
15     pthread_create(&id, NULL, foo, &j);
16     int* ptr;
17     pthread_join(id, (void**)&ptr);
18     printf("Value received by parent from child: ");
19     printf("%i\n", *ptr);
20 }
```

```
C:\Users\DELL\OneDrive\Documents\multithreading.exe
Value received as argument in starting routine: 1
Value received by parent from child: 2

-----
Process exited after 0.06445 seconds with return value 0
Press any key to continue . . .
```

```

1 #include<stdio.h>
2 #include<pthread.h>
3 #include<semaphore.h>
4 #include<unistd.h>
5
6 sem_t room;
7 sem_t chopstick[5];
8 void * philosopher(void *);
9 void eat(int);
10 int main()
11 {
12     int i,a[5];
13     pthread_t tid[5];
14     sem_init(&room,0,4);
15     for(i=0;i<5;i++){
16         sem_init(&chopstick[i],0,1);
17     }
18     for(i=0;i<5;i++){
19         a[i]=i;
20         pthread_create(&tid[i],NULL,philosopher,(void *)&a[i]);
21     }
22     for(i=0;i<5;i++){
23         pthread_join(tid[i],NULL);
24     }
25 void * philosopher(void * num)
26 {
27     int phil=*(int *)num;
28     sem_wait(&room);
29     printf("\nPhilosopher %d has entered room",phil);
30     sem_wait(&chopstick[phil]);
31     sem_wait(&chopstick[(phil+1)%5]);
32     eat(phil);
33     sleep(2);
34     printf("\nPhilosopher %d has finished eating",phil);
35     sem_post(&chopstick[(phil+1)%5]);
36     sem_post(&chopstick[phil]);
37     sem_post(&room);
38 }
39 void eat(int phil)
40 {
41     printf("\nPhilosopher %d is eating",phil);

```

sources Compile Log Debug Find Results Console

Col: 31 Sel: 0 Lines: 41 Length: 884 Insert

C:\Users\aswin\Documents\diningphilos.exe

```

Philosopher 0 has entered room
Philosopher 3 has entered room
Philosopher 2 has entered room
Philosopher 1 has entered room
Philosopher 0 is eating
Philosopher 3 is eating
Philosopher 3 has finished eating
Philosopher 0 has finished eating
Philosopher 2 is eating
Philosopher 4 has entered room
Philosopher 4 is eating
Philosopher 2 has finished eating
Philosopher 4 has finished eating
Philosopher 1 is eating
Philosopher 1 has finished eating

```

Process exited after 10.84 seconds with return value 0
Press any key to continue . . .

Done parsing in 0.174 seconds



Terminal

reenaasprgm.c x *program2.c x rkive.c x ipc.c x Untitled Document 2 x

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/shm.h>
#include<string.h>
int main()
{
    int i;
    void *shared_memory;
    char buff[100];
    int shmid;
    shmid=shmget((key_t)2345, 1024, 0666|IPC_CREAT);
    printf("Key of shared memory is %d\n",shmid);
    shared_memory=shmat(shmid,NULL,0);
    printf("Process attached at %p\n",shared_memory);
    printf("Enter some data to write to shared memory\n");
    read(0,buff,100);
    strcpy(shared_memory,buff);
    printf("You wrote : %s\n",(char *)shared_memory);
}
```

```
guest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF: ~/Desktop
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~$ cd Desktop
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ cc ipc.c
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ ./a.out
Key of shared memory is 2883597
Process attached at 0x7f2a03434000
Enter some data to write to shared memory
HELLO WORLD
You wrote : HELLO WORLD
**
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ cc ipc.c
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ ./a.out
Key of shared memory is 2883597
Process attached at 0x7faf86274000
Enter some data to write to shared memory
NAME OF THE STUDENT:ASWINI
You wrote : NAME OF THE STUDENT:ASWINI
**
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ cc ipc.c
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ ./a.out
Key of shared memory is 2883597
Process attached at 0x7fc9f0249000
Enter some data to write to shared memory
MY NAME IS REENA
You wrote : MY NAME IS REENA
***
quest-VU6MzW@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$
```