

```
/* fcfs scheduling(array) */
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main() {
```

```
    int i, n; //i+1=pid as i=0
```

```
    printf("\nFCFS scheduling..\n");
```

```
    printf("Enter the number of processes: ");
```

```
    scanf("%d", &n);
```

```
    int burst_time[n]; // Array to store burst times
```

```
    int waiting_time[n]; // Array to store waiting times
```

```
    int turnaround_time[n]; // Array to store turnaround times
```

```
    int totwtime = 0, totttime = 0;
```

```
    // if (n > 10) {
```

```
        // printf("Error: Maximum number of processes allowed is 10.\n");
```

```
        // return 1;
```

```
    // }
```

```
    for (i = 0; i < n; i++) {
```

```
        printf("\nEnter burst time for process %d: ", i + 1);
```

```
        scanf("%d", &burst_time[i]);
```

```
    }
```

```
    // Calculate waiting time and turnaround time
```

```
    waiting_time[0] = 0; //as it is the 1st process(no at here)
```

```
    turnaround_time[0] = burst_time[0]; //bt is always equal to tt for 1st process
```

```
    totttime = turnaround_time[0]; //total tt of 1st process=tt of 1st process
```

```
    for (i = 1; i < n; i++) {
```

```
        waiting_time[i] = waiting_time[i - 1] + burst_time[i - 1];
```

```

    turnaround_time[i] = waiting_time[i] + burst_time[i];

    totwtime += waiting_time[i];

    totttime += turnaround_time[i];
}

printf("\nProcesses\tBurst\tWaiting\tTurnaround\n");

for (i = 0; i < n; i++) {

    printf("%d\t%d\t%d\t%d\n", i + 1, burst_time[i], waiting_time[i], turnaround_time[i]);

}

printf("\nAverage Waiting time = %f", (float)totwtime / n);

printf("\nAverage Turnaround time = %f\n", (float)totttime / n);

return 0;
}

```

The screenshot shows a web browser window with the URL `tutorialspoint.com/compile_c_online.php`. The page title is "Online C Compiler". The code editor contains the C program for FCFS scheduling. The terminal output shows the program's execution results.

```

20- for (i = 0; i < n; i++) {
21-     printf("\nEnter burst time for process %d: ", i + 1);
22-     scanf("%d", &burst_time[i]);
23- }
24-
25- // calculate waiting time and turnaround time
26- waiting_time[0] = 0; //as it is the 1st process(no at here)
27- turnaround_time[0] = burst_time[0]; //bt is always equal to tt for 1st
   process
28- totttime = turnaround_time[0]; //total tt of 1st process=tt of 1st process
29-
30- for (i = 1; i < n; i++) {
31-     waiting_time[i] = waiting_time[i - 1] + burst_time[i - 1];
32-     turnaround_time[i] = waiting_time[i] + burst_time[i];
33-     totwtime += waiting_time[i];
34-     totttime += turnaround_time[i];
35- }
36-
37- printf("\nProcesses\tBurst\tWaiting\tTurnaround\n");
38- for (i = 0; i < n; i++) {
39-     printf("%d\t%d\t%d\t%d\n", i + 1, burst_time[i], waiting_time[i],
   turnaround_time[i]);
40- }
41-
42- printf("\nAverage Waiting time = %f", (float)totwtime / n);
43- printf("\nAverage Turnaround time = %f\n", (float)totttime / n);
44-
45- return 0;
46- }

```

Terminal Output:

```

FCFS scheduling..
Enter the number of processes: 3
Enter burst time for process 1: 5
Enter burst time for process 2: 10
Enter burst time for process 3: 15
Processes  Burst  Waiting Turnaround
1         5      0       5
2        10      5      15
3        15     15     30

Average Waiting time = 6.666667
Average Turnaround time = 16.666666

```