

C code to implement proportional scheduling

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

```
typedef struct {
    char name[5];
    int tickets;
} process;
```

```
int main() {
    int n, total_tickets = 0;
    float total_T = 0.0;
    printf("Enter no. of processes");
    scanf("%d", &n);
    process p[n];
    srand(time(NULL));
    for (int i = 0; i < n; i++) {
        printf("\n process %d\n", i+1);
        sprintf(p[i].name, "P%d", i+1);
        printf("Tickets");
        scanf("%d", &p[i].tickets);
        total_tickets = p[i].tickets;
        total_T += p[i].tickets;
    }
```

```
printf("\n --proportional share scheduling");
printf("Enter time period for scheduling: ");
int m;
scanf("%d", &m);
```



```

for (int i=0; i<m; i++) {
    int winning_ticket = rand() % total_tickets;
    int accumulated_tickets = 0;
    int winner_index;
    for (int j=0; j<n; j++) {
        accumulated_tickets += p[j].tickets;
        if (winning_ticket <= accumulated_tickets) {
            winner_index = j;
            break;
        }
    }

    printf("Tickets picked: %d, winner: %s\n",
           winning_ticket, p[winner_index].name);
}

for (int i=0; i<n; i++) {
    printf("\n the process: %s gets %.02f % of processor time\n",
           p[i].name, ((p[i].tickets / total_tickets) * 100));
}

return 0;
}

```

out put:

Enter number of processes: 3

Process 1:

Tickets: 10

Process 2:

Tickets: 20

Process 3:

Tickets: 30

proportional share scheduling: 5

Tickets picked: 5 winner: P1

Tickets picked: 25 winner: P2



Ticket picked: 35, winner: P<sub>3</sub>

Ticket picked: 50, winner: P<sub>3</sub>

Ticket picked: 15, winner: P<sub>2</sub>

The process: P<sub>1</sub> gets 16.67% of processor time.

The process: P<sub>2</sub> gets 33.33% of processor time

The process: P<sub>3</sub> gets 50.00% of processor time.

Ans  
6/11/24