praveenkumar.R
192324012

Experiment:4-Construct a scheduling program with C that selects the waiting process with the smallest execution time to execute next

Aim:

To implement the Shortest Job Next (SJN) CPU Scheduling algorithm.

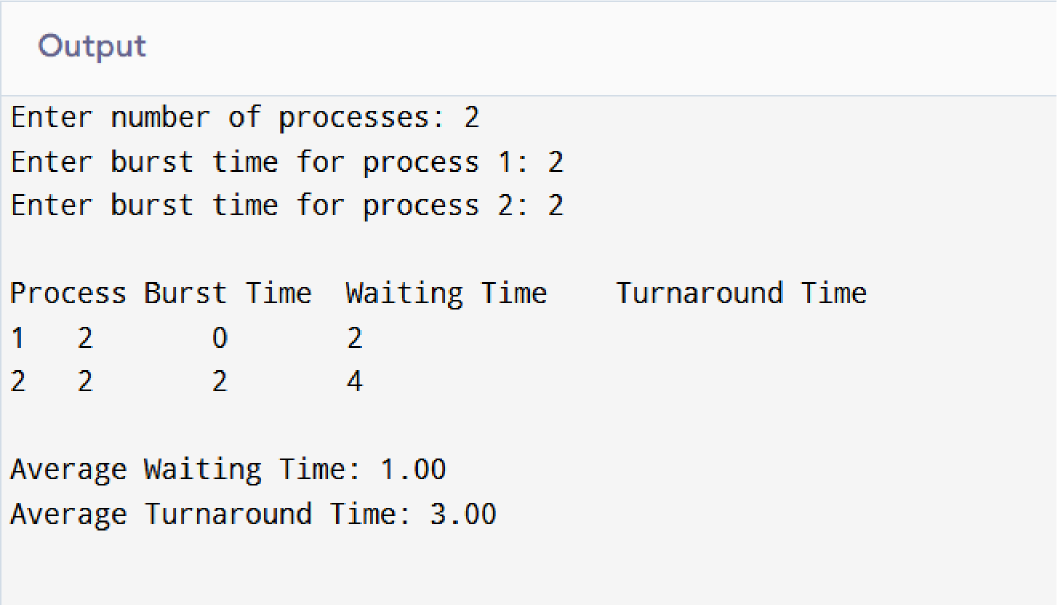
Procedure:

1. Sort processes based on their burst time in ascending order.
2. Calculate waiting time, turnaround time, and display the scheduling order.

C Program:

```
#include <stdio.h>

#include <stdlib.h>


struct Process {

   int id;

   int burst_time;

   int waiting_time;

   int turnaround_time;

};


int compare(const void *a, const void *b) {

   return ((struct Process *)a)->burst_time - ((struct Process *)b)->burst_time;

}


int main() {

   int n;

   printf("Enter number of processes: ");

   scanf("%d", &n);


   struct Process processes[n];

   int total_waiting_time = 0, total_turnaround_time = 0;

   for (int i = 0; i < n; i++) {
       processes[i].id = i + 1;
       printf("Enter burst time for process %d: ", i + 1);
       scanf("%d", &processes[i].burst_time);
   }
   qsort(processes, n, sizeof(struct Process), compare);
   processes[0].waiting_time = 0;
   processes[0].turnaround_time = processes[0].burst_time;
   total_turnaround_time = processes[0].turnaround_time;

   for (int i = 1; i < n; i++) {
       processes[i].waiting_time = processes[i - 1].waiting_time + processes[i - 1].burst_time;
       processes[i].turnaround_time = processes[i].waiting_time + processes[i].burst_time;
       total_waiting_time += processes[i].waiting_time;
       total_turnaround_time += processes[i].turnaround_time;
   }
   printf("\nProcess\tBurst Time\tWaiting Time\tTurnaround Time\n");
   for (int i = 0; i < n; i++) {
       printf("%d\t%d\t\t%d\t\t%d\n", processes[i].id, processes[i].burst_time,
processes[i].waiting_time, processes[i].turnaround_time);
   }

   printf("\nAverage Waiting Time: %.2f\n", (float)total_waiting_time / n);
   printf("Average Turnaround Time: %.2f\n", (float)total_turnaround_time / n);

   return 0;
}
```

Output:



```
Enter number of processes: 2
Enter burst time for process 1: 2
Enter burst time for process 2: 2

Process Burst Time  Waiting Time    Turnaround Time
1   2       0       2
2   2       2       4

Average Waiting Time: 1.00
Average Turnaround Time: 3.00
```