```
#include<stdio.h>
struct priority_scheduling
{
  char process_name;
  int burst_time,waiting_time,turn_around_time,priority;
};
int main()
{
  int number_of_process,total = 0,ASCII_number = 65,position,i,j;
  struct priority_scheduling temp_process;
  float average_waiting_time;
  float average_turnaround_time;
  printf("Enter the total number of Processes: ");
  scanf("%d", & number_of_process);
  struct priority_scheduling process[number_of_process];
  printf("\nPlease Enter the  Burst Time and Priority of each process:\n");
  for (i = 0; i < number_of_process; i++)
  {
    process[i].process_name = (char) ASCII_number;
    printf("\nEnter the details of the process %c \n", process[i].process_name);
    printf("Enter the burst time: ");
    scanf("%d", & process[i].burst_time);
    printf("Enter the priority: ");
    scanf("%d", & process[i].priority);
    ASCII_number++;
  }
  for (i = 0; i < number_of_process; i++)
  {
    position = i;
    for (j = i + 1; j < number_of_process; j++)
            {
```

```
    if (process[j].priority > process[position].priority)
      position = j;
   }
   temp_process = process[i];
   process[i] = process[position];
   process[position] = temp_process;
  }
  process[0].waiting_time = 0;
  for (i = 1; i < number_of_process; i++)
  {
   process[i].waiting_time = 0;
   for (j = 0; j < i; j++) {
    process[i].waiting_time += process[j].burst_time;
   }
   total += process[i].waiting_time;
  }
  average_waiting_time = (float) total / (float) number_of_process;
  total = 0;
  printf("\n\nProcess_name \t Burst Time \t Waiting Time \t  Turnaround Time\n");
  printf("-----------------------------------------------------------------\n");
  for (i = 0; i < number_of_process; i++)
  {
   process[i].turn_around_time = process[i].burst_time + process[i].waiting_time;
   total += process[i].turn_around_time;
   printf("\t  %c \t\t  %d \t\t %d \t\t %d", process[i].process_name, process[i].burst_time,
process[i].waiting_time, process[i].turn_around_time);
   printf("\n-----------------------------------------------------------------\n");
  }
  average_turnaround_time = (float) total / (float) number_of_process;
  printf("\n\n Average Waiting Time : %f", average_waiting_time);
  printf("\n Average Turnaround Time: %f\n", average_turnaround_time);
```

```
  return 0;
}
```