Practical-12

Aim: To show priority scheduling in C/C++.

In priority scheduling, every process is associated with a priority ranging from 0-10 where, integer 0 represents the lowest priority and 10 represents the highest priority. Priorities can be defined in two ways i.e. internally and externally. Also, priority scheduling can be either preemptive or nonpreemptive.

Code:

```
#include <stdio.h>
void swap(int *a,int *b)
{
   int temp=*a;
   *a=*b;
   *b=temp;
}
int main()
{
   int n;
   printf("Enter Number of Processes: ");
   scanf("%d",&n);
   int b[n],p[n],index[n];
   for(int i=0;i<n;i++)
   {
      printf("Enter Burst Time for Process %d: ",i+1);
      scanf("%d",&b[i]);
      printf("Enter Priority Value for proccess %d:",i+1);
      scanf("%d",&p[i]);
      index[i]=i+1;
   }
   for(int i=0;i<n;i++)
   {
      int a=p[i],m=i;
      for(int j=i;j<n;j++)
      {
```

```
            if(p[j] < a)
            {
                a=p[j];
                m=j;
            }
        }
        swap(&p[i], &p[m]);
        swap(&b[i], &b[m]);
        swap(&index[i],&index[m]);
    }
    int t=0;
    printf("Order of process Execution is\n");
    for(int i=0;i<n;i++)
    {
        printf("P%d is executed from %d to %d\n",index[i],t,t+b[i]);
        t+=b[i];
    }
    printf("\n");
    printf("Process Id     Burst Time   Wait Time    TurnAround Time\n");
    int wait_time=0;
    for(int i=0;i<n;i++)
    {
        printf("P%d          %d          %d          %d\n",index[i],b[i],wait_time,wait_time +
b[i]);
        wait_time += b[i];
    }
    return 0;
}
```

```
C:\Users\plumb\CLionProjects\untitled\cmake-build-debug\u
Enter Number of Processes:3
 Enter Burst Time for Process 1:2
 Enter Priority Value for proccess 1:2
Enter Burst Time for Proc
ess 2:3
 Enter Priority Value for proccess 2:3
Enter Burst Time for Process 3:4
 Enter Priority Value for proccess 3:4
Order of
process Execution is
P1 is executed from 0 to 2
P2 is executed from 2 to 5
P3 is executed from 5 to 9

Process Id     Burst Time   Wait Time    TurnAround Time
P1             2              0             2
P2             3              2             5
P3             4              5             9

Process finished with exit code 0
```