Ex. No.: 6c)
Date: 05/02/25

PRIORITY SCHEDULING

Aim:

To implement priority scheduling technique

Algorithm:

1. Get the number of processes from the user.
2. Read the process name, burst time and priority of process.
3. Sort based on burst time of all processes in ascending order based priority 4.
Calculate the total waiting time and total turnaround time for each process 5.
Display the process name & burst time for each process.
6. Display the total waiting time, average waiting time, turnaround time

Program Code:

```c
#include <stdio.h>
int main(){
    int n;
    printf("Enter no. of processes");
    scanf("%d", &n);
    int p[n], bt[n], wt[n], tat[n], pr[n], t1, t2, t3;
    float sum1=0, sum2=0;
    printf("enter process no. Bt of priority \n");
    for(int i=0; i<n; i++){
        scanf("%d %d %d", &p[i], &bt[i], &pr[i]);
    }
    for(int i=0; i<n-1; i++){
        for(int j=0; j<n-1-i; j++){
            if(pr[j] > pr[j+1]){
                t1 = pr[j];
                pr[j] = pr[j+1];
                pr[j+1] = t1;
```

```c
            t₂ = P[j];
            P[j] = P[j+1];
            P[j+1] = t2;

                t3  =  bt [j];
                bt [j] = bt [j+1];
                bt[j+1] = t3;
            }
        }
    }

wt [0] = 0;
 For (int i=1 ; i<n; i++) {
        wt [i] = wt [i-1] + bt [i-1];
        s₁ = wt [i];
    }
 for (int i=0 ; i<n ; i++){
        tat [i] =  wt [i] + bt [i];
        S₂ + = tat [i];
    }
 for (int i=0 ; i<n; i++){
        Printf (" %d lt %d lt %d lt %d lt %d ln",
                        P[i] , bt [i], pr[i], wt [i], tat[i]);

    }

    Printf ("\n avg  wt = %.2f," s₁/n);
    Printf ("\n avg  tat = %.2f," s₂/n);
}
```

Output

enter n
4
enter process no, Bt, Priority

1	8	2
2	4	1
3	6	4
4	3	3

Process	BT	Priority	WT	TAT
P_2	4	1	0	4
P_1	8	2	4	12
P_4	3	3	12	15
P_3	6	4	15	21

avg WT = 7.75 ms
avg tat = 13.00 ms

Sample Output:

```
C:\Users\admin\Desktop\Untitled1.exe

Enter Total Number of Process:4

Enter Burst Time and Priority

P[1]
Burst Time:6
Priority:3

P[2]
Burst Time:2
Priority:2

P[3]
Burst Time:14
Priority:1

P[4]
Burst Time:6
Priority:4

Process        Burst Time          Waiting Time      Turnaround Time
P[3]               14                   0                   14
P[2]               2                    14                  16
P[1]               6                    16                  22
P[4]               6                    22                  28

Average Waiting Time-13
Average Turnaround Time-20
```

Result:

Hence the priority scheduling is implemented and executed successfully.

43