

Ex. No.: 6c)

Date: 13/12/25
08

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.
4. Calculate the total waiting time and total turnaround time for each process 5.
5. Display the process name & burst time for each process.
6. Display the total waiting time, average waiting time, turnaround time

Program Code:

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

```
int main()
```

```
{
```

```
    int n, i, j;
```

```
    printf("Enter the number of process:");
```

```
    scanf("%d", &n);
```

```
    int bt[n], wt[n], tat[n], p[n], pri[n], kump;
```

```
    float total_wt = 0, total_tat = 0;
```

```
    printf("Enter the burst time and priority: \n");
```

```
    for (i = 0; i < n; i++) {
```

```
        printf("P[%d]", i);
```

```
        printf("Burst time:");
```

```
        scanf("%d", &bt[i]);
```

```
        printf("Priority:");
```

```
        p[i] = i;
```

```
        scanf("%d", &pri[i]);
```

```
}
```

```

for (i=0; i<n-1; i++) {
    for (j=i+1; j<n; j++) {
        if (pri[i] > pri[j]) {
            temp = bt[i];
            bt[i] = bt[j];
            bt[j] = temp;
            temp = pri[i];
            pri[i] = pri[j];
            pri[j] = temp;
            temp = p[i];
            p[i] = p[j];
            p[j] = temp;
        }
    }
}

```

wt[0] = 0;

```

for (i=1; i<n; i++) {
    wt[i] = wt[i-1] + bt[i-1];
}

```

```

for (i=0; i<n; i++) {
    tat[i] = bt[i] + wt[i];
}

```

printf("\n Process \t Burst time \t Waiting time \t Turnaround time \n");

```

for (i=0; i<n; i++) {
    printf("%d \t %d \t %d \t %d", p[i],
           bt[i], wt[i], tat[i]);
}

```

total_wt += wt[i];

total_tat += tat[i];

printf("\n Average waiting time %.1f \n", total_wt/n);

printf("\n Average ~~turnaround~~ time %.1f \n", total_tat/n);

}

Output

Enter the total no of process : 4

Enter Burst time and priority

P[1]

Burst time : 6

Priority : 2

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

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 above priority scheduling is executed successfully.

[Handwritten signature]