Atma Ram Sanatan Dharma College
(University Of Delhi)

Name Neeraj
Roll no 18088
Course B.Sc (H) Computer science
Submitted To Dr. Parul Jain

Q 9. Write a program to implement non-preemptive priority based scheduling algorithm?

Answer

```
#include<stdio.h>

int main()
{
	int bt[20],p[20],wt[20],tat[20],pr[20],i,j,n,total=0,pos,temp;
	float avg_wt,avg_tat;

	printf("\nEnter total number of process: ");
	scanf("%d",&n);

	printf("\nEnter Burst time and priority: \n");
	for(i=0;i<n;i++)
	{
		printf("\nProcess %d ->", i+1);
		printf("Burst time: ");
```

```
        scanf("%d", &bt[i]);
        printf("Priority: ");
        scanf("%d", &pr[i]);
        p[i] = i+1;
}

for(i=0; i<n; i++)
{
        pos=i;
        for(j=i+1; j<n; j++)
        {
        if(pr[j]<pr[pos])
           pos = j;
        }

        temp = pr[i];
        pr[i] = pr[pos];
        pr[pos] = temp;

        temp = bt[i];
        bt[i] = bt[pos];
        bt[pos] = temp;

        temp  = p[i];
        p[i] = p[pos];
        p[pos] = temp;
        }

        wt[0] = 0;
        for(i=1;i<n;i++)
        {
        wt[i] = 0;
```

```
            for(j=0; j<i; j++)
                wt[i] += bt[i];
            total+=wt[i];
        }

        avg_wt = (float) total/n;
        total = 0;

        printf("\nProcess \tBurst Time \t Wating Time \tTurn Around Time");
        for(i=0;i<n;i++)
        {
            tat[i] = bt[i] + wt[i];
            total += tat[i];
            printf("\nP: %d \t\t%d \t\t\t\t%d \t\t%d", p[i],bt[i],wt[i],tat[i]);
        }

        avg_tat = (float)total/n;

        printf("\n\nAverage wating time: %f",avg_wt);
        printf("\nAverage trun around time: %f\n",avg_tat);
        return 0;
}
```

Output:-

```
gautam@gautam:~/Desktop/os$ gcc -o a program11.c
gautam@gautam:~/Desktop/os$ ./a

Enter total number of process: 3

Enter Burst time and priority:

Process 1 ->Burst time: 2
Priority: 1

Process 2 ->Burst time: 4
Priority: 2

Process 3 ->Burst time: 3
Priority: 3

Process         Burst Time       Wating Time     Turn Around Time
P: 1            2                                0               2
P: 2            4                                4               8
P: 3            3                                6               9

Average wating time: 3.333333
Average trun around time: 6.333333
```

Q 10. Write a program to implement preemptive priority based scheduling algorithm?

Answer

```
#include<stdio.h>

int main()
{
	int i,j,n,time,sum_wait=0,sum_turnaround=0,smallest;
	int at[10],bt[10],pt[10],rt[10],remain;
	printf("\nEnter number of processes: ");
	scanf("%d", &n);
	remain = n;

	printf("\nEnter arrival time, burst time and priority for: \n");
	for(i=0; i<n; i++)
	{
		printf("\nProcess %d: ",i+1);
		scanf("%d", &at[i]);
		scanf("%d", &bt[i]);
		scanf("%d", &pt[i]);
		rt[i] = bt[i];
	}

	pt[9] = 11;
	printf("\n\nProcess \t|TurnAround time |Waiting time\n");
	for(time=0; remain != 0; time++)
	{
		smallest = 9;
		for(i=0; i<n; i++)
		{
```

```
            if(at[i] <= time && pt[i] <pt[smallest] && rt[i] >0)
            {
            smallest = i;
            }
        }

        rt[smallest]--;
        if(rt[smallest] == 0)
        {
            remain--;
            printf("\nP: %d \t|\t %d \t|\t %d\n", smallest+1,
time+1-at[smallest], time+1-at[smallest]-bt[smallest]);
            sum_wait+=time+1-at[smallest];
            sum_turnaround+=time+1-at[smallest]-bt[smallest];
        }
    }

    printf("\nAverage waiting time: %f",sum_wait*1.0/n);
    printf("\n\nAverage Trun Around Time: %f\
n",sum_turnaround*1.0/n);
    return 0;
}
```

Output:-

```
gautam@gautam:~/Desktop/os$ gcc -o a program10.c
gautam@gautam:~/Desktop/os$ ./a

Enter number of processes: 4

Enter arrival time, burst time and priority for:

Process 1: 0 4 1

Process 2: 1 3 2

Process 3: 4 5 3

Process 4: 6 3 4

Process          |TurnAround time |Waiting time

P: 1     |         4       |         0

P: 2     |         6       |         3

P: 3     |         8       |         3

P: 4     |         9       |         6

Average waiting time: 6.750000

Average Trun Around Time: 3.000000
```