

OPERATING SYSTEM - CS23431

EXP 6(D)

ROUND ROBIN SCHEDULING

NAME: KRITHIKA B

ROLL NO: 230701156

PROGRAM:

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter number of processes: ");
```

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

```
    int p[n], a[n], bt[n], temptbt[n], slot;
```

```
    printf("Enter process ID, arrival time, burst time for each process:\n");
```

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

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

```
        temptbt[i] = bt[i];
```

```
    }
```

```
    printf("Enter quantum time slot: ");
```

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

```
    int totalwt = 0, totalturn = 0, totaltime = 0;
```

```
    int i = 0, count = 0, completed = 0;
```

```
    printf("P_ID\tBT\tTAT\tWT\n");
```

```

while (completed != n) {
    if (temptbt[i] <= slot && temptbt[i] > 0) {
        totaltime += temptbt[i];
        temptbt[i] = 0;
        count = 1;
    }
    else if (temptbt[i] > 0) {
        totaltime += slot;
        temptbt[i] -= slot;
    }

    if (temptbt[i] == 0 && count == 1) {
        completed++;
        int tat = totaltime - a[i];
        int wt = totaltime - a[i] - bt[i];
        printf("%d\t%d\t%d\t%d\n", p[i], bt[i], tat, wt);
        totalwt += wt;
        totalturn += tat;
        count = 0;
    }

    if (i == n - 1)
        i = 0;
    else
        i++;
}

```

```
printf("Average waiting time is %d\n", totalwt / n);  
printf("Average turn around time is %d\n", totalturn / n);  
  
return 0;  
}
```

OUTPUT:

```
Enter number of processes: 4  
Enter process ID, arrival time, burst time for each process:  
1 0 4  
2 1 7  
3 2 5  
4 3 6  
Enter quantum time slot: 3  
P_ID    BT    TAT    WT  
1       4     13     9  
3       5     16     11  
4       6     18     12  
2       7     21     14  
Average waiting time is 11  
Average turn around time is 17  
[cse164@fedora ~]$
```