

C Program:

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

void main()
{
    // initialize the variable name
    int i, NOP, sum=0, count=0, y, quant, wt=0, tat=0, at[10], bt[10], temp[10];
    float avg_wt, avg_tat;
    printf(" Total number of process in the system: ");
    scanf("%d", &NOP);
    y = NOP; // Assign the number of process to variable y

    // Use for loop to enter the details of the process like Arrival time and the Burst Time
    for(i=0; i<NOP; i++)
    {
        printf("\n Enter the Arrival and Burst time of the Process[%d]\n", i+1);
        printf(" Arrival time is: \t"); // Accept arrival time
        scanf("%d", &at[i]);
        printf(" \nBurst time is: \t"); // Accept the Burst time
        scanf("%d", &bt[i]);
        temp[i] = bt[i]; // store the burst time in temp array
    }
    // Accept the Time quantat
    printf("Enter the Time Quantum for the process: \t");
    scanf("%d", &quant);
    // Display the process No, burst time, Turn Around Time and the waiting time
    printf("\n Process No \t\t Burst Time \t\t TAT \t\t Waiting Time ");
    for(sum=0, i = 0; y!=0; )
    {
        if(temp[i] <= quant && temp[i] > 0) // define the conditions
        {
            sum = sum + temp[i];
            temp[i] = 0;
            count=1;
        }
    }
}
```

```

    }
else if(temp[i] > 0)
{
    temp[i] = temp[i] - quant;sum =
    sum + quant;
}
if(temp[i]==0 && count==1)
{
    y--; //decrement the process no.
    printf("\nProcess No[%d] \t\t %d\t\t\t\t %d\t\t\t %d", i+1, bt[i], sum-at[i], sum-at[i]-bt[i]);
    wt = wt+sum-at[i]-bt[i];tat =
    tat+sum-at[i]; count =0;
}
if(i==NOP-1)
{
    i=0;
}
else if(at[i+1]<=sum)
{
    i++;
}
else
{
    i=0;
}
}
// represents the average waiting time and Turn Around time

```

5.0 Output of Micro-project:

```
Total number of process in the system: 5

Enter the Arrival and Burst time of the Process[1]
Arrival time is:      0

Burst time is: 6

Enter the Arrival and Burst time of the Process[2]
Arrival time is:      0

Burst time is: 5

Enter the Arrival and Burst time of the Process[3]
Arrival time is:      0

Burst time is: 2

Enter the Arrival and Burst time of the Process[4]
Arrival time is:      0

Burst time is: 3

Enter the Arrival and Burst time of the Process[5]
Arrival time is:      0

Burst time is: 7
Enter the Time Quantum for the process:      2

Process No      Burst Time      TAT      Waiting Time
Process No[3]    2      6      4
Process No[4]    3      15     12
Process No[1]    6      19     13
Process No[2]    5      20     15
Process No[5]    7      23     16
Average Turn Around Time:      12.000000
Average Waiting Time: 16.600000

...Program finished with exit code 0
Press ENTER to exit console.
```