

OS ALGORITHM

Cb.en.u4cys21047

```
1.#include<stdio.h>
int main()
{
    int bt[20],p[20],wt[20],tat[20],i,j,n,total=0,pos,temp;
    float avg_wt,avg_tat;
    printf("Enter number of process:");
    scanf("%d",&n);

    printf("\nEnter Burst Time:n");
    for(i=0;i<n;i++)
    {
        printf("p%d:\n",i+1);
        scanf("%d",&bt[i]);
        p[i]=i+1;
    }

    //sorting of burst times
    for(i=0;i<n;i++)
    {
        pos=i;
        for(j=i+1;j<n;j++)
        {
            if(bt[j]<bt[pos])
                pos=j;
        }

        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[j];

    total+=wt[i];
}

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

printf("\nProcess\tBurst Time\tWaitTime\tTurnaround Time");
for(i=0;i<n;i++)
{
    tat[i]=bt[i]+wt[i];
    total+=tat[i];
    printf("\np%d\t\t %d\t\t %d\t\t%d",p[i],bt[i],wt[i],tat[i]);
}

avg_tat=(float)total/n;
printf("\n\nAverage Waiting Time=%f",avg_wt);
printf("\nAverage Turnaround Time=%f\n",avg_tat);
}

```

```

Last login: Thu Dec  1 09:58:35 on ttys000
(base) vihalroy@Ns-MacBook-Pro ~ % gcc os.c
(base) vihalroy@Ns-MacBook-Pro ~ % ./a.out
Enter number of process: 3
Enter Burst Time:
P1: 3
P2: 3
P3: 2

```

P	BT	WT	TAT
P3	2	0	2
P2	3	2	5
P1	3	5	8

```

Average Waiting Time= 2.333333
Average Turnaround Time= 5.000000%
(base) vihalroy@Ns-MacBook-Pro ~ %

```

```

2.#include <stdio.h>
int main()
{
    int pid[15];
    int bt[15];
    int n;
    printf("Enter the number of processes: ");
    scanf("%d",&n);

    printf("Enter process id of all the processes: ");
    for(int i=0;i<n;i++)
    {
        scanf("%d",&pid[i]);
    }

    printf("Enter burst time of all the processes: ");
    for(int i=0;i<n;i++)
    {
        scanf("%d",&bt[i]);
    }

    int i, wt[n];
    wt[0]=0;

    //for calculating waiting time of each process
    for(i=1; i<n; i++)
    {
        wt[i]= bt[i-1]+ wt[i-1];
    }

    float twt=0.0;
    float tat= 0.0;
    for(i=0; i<n; i++)
    {
        printf("Process:%d\n", pid[i]);

        printf("burst time:%d\n", bt[i]);
        printf("waiting time:%d\n", wt[i]);
    }
}

```

```

//calculating and printing turnaround time of each process
printf("turnaround time:%d\n", bt[i]+wt[i]);
printf("\n");

//for calculating total waiting time
twl += wt[i];

//for calculating total turnaround time
tat += (wt[i]+bt[i]);
}
float att,awt;

//for calculating average waiting time
awt = twl/n;

//for calculating average turnaround time
att = tat/n;
printf("Avg. waiting time= %f\n",awt);
printf("Avg. turnaround time= %f",att);
}

```

```

vihalroy — -zsh — 80x24
[(base) vihalroy@Ns-MacBook-Pro ~ % gcc 123os.c
[(base) vihalroy@Ns-MacBook-Pro ~ % ./a.out
Enter number of process:3

Enter Burst Time:np1:
1
p2:
3
p3:
4

Process Burst Time      WaitTime      Turnaround Time
p1                1                0                1
p2                3                1                4
p3                4                4                8

Average Waiting Time=1.666667
Average Turnaround Time=4.333333
(base) vihalroy@Ns-MacBook-Pro ~ %

```

```

3.#include <stdio.h>
int main()
{
    int A[100][4]; // Matrix for storing Process Id, Burst
                  // Time, Average Waiting Time & Average
                  // Turn Around Time.

    int i, j, n, total = 0, index, temp;
    float avg_wt, avg_tat;
    printf("Enter number of process: ");
    scanf("%d", &n);
    printf("Enter Burst Time:\n");
    // User Input Burst Time and allotting Process Id.
    for (i = 0; i < n; i++) {
        printf("P%d: ", i + 1);
        scanf("%d", &A[i][1]);
        A[i][0] = i + 1;
    }
    // Sorting process according to their Burst Time.
    for (i = 0; i < n; i++) {
        index = i;
        for (j = i + 1; j < n; j++)
            if (A[j][1] < A[index][1])
                index = j;
        temp = A[i][1];
        A[i][1] = A[index][1];
        A[index][1] = temp;

        temp = A[i][0];
        A[i][0] = A[index][0];
        A[index][0] = temp;
    }
    A[0][2] = 0;
    // Calculation of Waiting Times
    for (i = 1; i < n; i++) {
        A[i][2] = 0;
        for (j = 0; j < i; j++)
            A[i][2] += A[j][1];
        total += A[i][2];
    }
    avg_wt = (float)total / n;

```

```

total = 0;
printf("P   BT   WT   TAT\n");
// Calculation of Turn Around Time and printing the
// data.
for (i = 0; i < n; i++) {
    A[i][3] = A[i][1] + A[i][2];
    total += A[i][3];
    printf("P%d   %d %d %d\n", A[i][0],
           A[i][1], A[i][2], A[i][3]);
}
avg_tat = (float)total / n;
printf("Average Waiting Time= %f",avg_wt);
printf("\nAverage Turnaround Time= %f",avg_tat);
}

```

```

vihalroy — -zsh — 80x24
(base) vihalroy@Ns-MacBook-Pro ~ % nano 234os.c
(base) vihalroy@Ns-MacBook-Pro ~ % gcc 234os.c
(base) vihalroy@Ns-MacBook-Pro ~ % ./a.out
Enter number of process: 3
Enter Burst Time:
P1: 1
P2: 1
P3: 2
P      BT      WT      TAT
P1     1       0       1
P2     1       1       2
P3     2       2       4
Average Waiting Time= 1.000000
Average Turnaround Time= 2.333333
(base) vihalroy@Ns-MacBook-Pro ~ %

```