```c
#include <stdio.h>
 int waitingtime(int proc[], int n, int burst_time[], int
wait_time[]) {    wait_time[0] = 0;
   for (int i = 1; i < n ; i++ )    wait_time[i] =
burst_time[i-1] + wait_time[i-1] ;    return 0;
}
int turnaroundtime( int proc[], int n, int burst_time[],
int wait_time[], int tat[]) {
   int i;
   for ( i = 0; i < n ; i++)    tat[i] = burst_time[i] +
wait_time[i];    return 0;
}
int avgtime( int proc[], int n, int burst_time[]) {    int
wait_time[n], tat[n], total_wt = 0, total_tat = 0;
   int i;
   waitingtime(proc, n, burst_time, wait_time);
turnaroundtime(proc, n, burst_time, wait_time, tat);
printf("Processes  Burst   Waiting Turn around \n");
   for ( i=0; i<n; i++) {       total_wt = total_wt +
wait_time[i];       total_tat = total_tat + tat[i];
printf(" %d\t  %d\t\t %d \t%d\n", i+1, burst_time[i],
wait_time[i], tat[i]);    }
   printf("Average waiting time = %f\n", (float)total_wt /
(float)n);    printf("Average turn around time = %f\n",
(float)total_tat / (float)n);    return 0;
}
int main() {    int proc[] = { 1, 2, 3};    int n = sizeof
proc / sizeof proc[0];    int burst_time[] = {5, 8, 12};
avgtime(proc, n, burst_time);
   return 0;
}
```

C

Demo.c

CODE OUTPUT

```
Processes  Burst   Waiting Turn around
 1   5   0  5
 2   8   5  13
 3   12   13  25
Average waiting time = 6.000000
Average turn around time = 14.333333
```

Date
Page No

Deepika Rawat
BSc-IT 2A
Roll No - 20052033
Campus - Hill campus.

Algorithm

START
Set wait_time [0] = 0
Loop for i = 1 and i < n and i++
Set wait_time [i] = burst_time [i-1] + wait_time [i-1]
End for
~~In function int turnaroundtime (int proc[], int n, int burst_time[], int wait_time [], int ta~~
STEP-1 In function int waitingtime (int proc [], int n, int burst_time [], int wait_time [])
STEP-2 In function int turnaroundtime (int proc [], int n, int burst_time [], int wait_time [], int tat [])
Loop for i = 0 and i < n and i++
Set tat [i] = burst_time [i] + wait_time [i]
End for
STEP-3 In function int avgtime (int proc [], int n, int burst_time [])
Declare and initialize wait_time [n], tat [n] total_wt=0
total_tat = 0;
Call waitingtime (proc, n, burst_time, waiting_time)
Call turnaroundtime (proc, n, burst_time, waitime, tat)
Loop for i=0 and i<n and i++
Set total_wt = total_wt + wait_time [i]
Set total_tat = total_tat + tat [i]
Print process number, burstime wait time and turnaround time
End for
Print " Average waiting time = i.e. total_wt /n
print " Average turn around time = i.e. total_tat /n

Date
Page No.

STEP-4 In int main ()
Declare the input int proc[] = {1,2,3}
Declare and initialize n = size of proc / size of proc[0]
Declare and initialize burst-time [] = {10,5,8}
Call avgtime (proc, n, burst-time)
Stop.