```cpp
#include<stdio.h>
#include<iostream>
using namespace std;


struct proc
{
    int pid;
    int at,bt,wt,tat,rbt;
    int flag,flag1;
};
struct proc p1[10];
int i,j,k,n,no,m;
float atat=0.0,awt=0.0;
int tbt=0;
int minimum1();
int main()
{
    int minv,locv,mins,locs;
    printf("\nNumber of processes\t:\t");
    scanf("%d",&n);
    printf("\nEnter the proc information\t:\n");
    printf("\npID (enter) Arrival Time (enter) Burst Time (Enter)\t:\t");
    for(i=0;i<n;i++)
    {
        p1[i].wt=0;
        p1[i].tat=0;
        p1[i].flag=0;
        p1[i].flag1=0;
        scanf("%d%d%d",&p1[i].pid,&p1[i].at,&p1[i].bt);
        tbt+=p1[i].bt;
        p1[i].rbt=p1[i].bt;
    }
    printf("\nThe Process information:");
    printf("\npID (enter)\tArrival Time (enter)\tBurst Time (Enter)");
    for(i=0;i<n;i++)
    {
        printf("\n%d\t%d\t%d\n",p1[i].pid,p1[i].at,p1[i].bt);
    }
    minv=p1[0].at;
    locv=0;
    for(i=1;i<n;i++)
    {
        if(p1[i].at<minv)
        {
            locv=i; //tells min at process in locv
            minv=p1[i].at;
        }
    }
    for(i=0;i<n;i++)
    {
        if(p1[i].at==minv)
        {
            p1[i].flag1=1; //processes having same minimum at
        }
    }
    mins=p1[0].bt;
    locs=0;
    for(i=0;i<n;i++)
    {
        if(p1[i].flag1==1&&p1[i].bt<mins)
        {
            mins=p1[i].bt; //gives process with minimum burst time
            locs=i;
        }
    }
    printf("\nGantt Chart:");
```

```
    for(i=minv;i<tbt+minv;i++)
    {
        for(j=0;j<n;j++)
        {
            if(p1[j].rbt>0&&p1[j].at<=i)
            {
                p1[j].flag=1;
            }
        }
        no=minimum1();
        printf("%d p[%d]",i,p1[no].pid);
        p1[no].rbt=p1[no].rbt-1;
        for(k=0;k<n;k++)
        {
            if(p1[k].rbt>0&&p1[k].at<=i&&k!=no)
            {
                p1[k].wt++;
            }
        }
    }
    printf("%d",tbt+minv);
    for(i=0;i<n;i++)
    {
        awt+=p1[i].wt;
    }
    awt=awt/n;
    for(i=0;i<n;i++)
    {
        p1[i].tat=p1[i].wt+p1[i].bt;
        atat+=p1[i].tat;
    }
    atat=atat/n;
    printf("\nAvg. Waiting Time = %f, Avg. Turn Around Time = %f",awt,atat);
    printf("\nThe Process information:");
    printf("\npID\tArrival Time\tBurst Time\tWaiting Time\tTurn Around Time");
    for(i=0;i<n;i++)
    {
        printf("\n%d\t\t%d\t\t%d\t\t%d\t\t%d\n",p1[i].pid,p1[i].at,p1[i].bt,p1[i].wt,p1[i].tat);
    }
    return 0;
}
int minimum1()
{
    int loc,z;
    int mini;
    mini=99;
    loc=-1;
    for(z=0;z<n;z++)
    {
        if(p1[z].rbt>0&&p1[z].at<=i&&p1[z].rbt<mini)
        {
            mini=p1[z].rbt;
            loc=z;
        }
    }
    return loc;
}
```