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

void RoundRob(int no,int remt[10],int Cur_t,int AT[10], int BT[10]);
main()
{
	int Pno,j,no,CurT,RemProc,indicator,TQ,WT,TAT,AT[10],BT[10],remt[10],x=1;
	indicator = 0;
	WT = 0;
	TAT = 0;
	printf("Enter number of processes ");
	scanf("%d",&no);
	RemProc = no;

	printf("\nEnter the arrival time and burst time of the processes\n");
	for(Pno = 0;Pno < no;Pno++)
	{
		printf("\nProcess P%d\n",Pno+1);
		printf("Arrival time = ");
		scanf("%d",&AT[Pno]);
		printf("Burst time = ");
		scanf("%d",&BT[Pno]);
		remt[Pno]=BT[Pno];
	}
	printf("The details of time quantum are as follows:\n");
	printf("The time quantum for first round is 3.\n");
	TQ=3;
	CurT=0;
	for(Pno=0;RemProc!=0;)
```

```
{
        if(remt[Pno]<=TQ && remt[Pno]>0)
        {
                CurT+=remt[Pno];
                remt[Pno]=0;
                indicator=1;
        }
        else if(remt[Pno]>0)
        {
                remt[Pno]-=TQ;
                CurT+=TQ;
        }
        if(remt[Pno]==0 && indicator==1)
        { printf("%d",Pno);
                RemProc--;
                printf("P %d",Pno+1);
                printf("\t\t\t%d",CurT-AT[Pno]);
                printf("\t\t\t%d\n",CurT-BT[Pno]-AT[Pno]);
                WT+=CurT-AT[Pno]-BT[Pno];
                TAT+=CurT-AT[Pno];
                indicator=0;

        }
        if(Pno==no-1){
                x++;
                if(x==2){
                        Pno=0;
                        TQ=6;
```

```
                        printf("The time quantum for second round is 6. \n");
                }
                else{
                        break;
                }
        }
        else if(CurT >= AT[Pno+1]){
                Pno++;
        }
        else{
                Pno=0;
        }
    }

    RoundRob(no,remt,CurT,AT,BT);

    return 0;
}


void RoundRob(int no,int remt[10],int Cur_t,int AT[10], int BT[10])
{

    float avg_wait,avg_tat;
  int i,j,n=no,temp,btime[20],Proc_no[20],w_time[20],tat_t[20],total=0,loc;

  printf("Third round with least burst time.\n");

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

```
{
   btime[i]=remt[i];
   w_time[i]=Cur_t-AT[i]-btime[i];
                    Proc_no[i]=i+1;
}

for(i=0;i<n;i++)
{
   loc=i;
   for(j=i+1;j<n;j++)
   {
      if(btime[j]<btime[loc]){
        loc=j;
      }
   }
   temp=btime[i];
   btime[i]=btime[loc];
   btime[loc]=temp;
   temp=Proc_no[i];
   Proc_no[i]=Proc_no[loc];
   Proc_no[loc]=temp;
}

for(i=1;i<n;i++)
{
   for(j=0;j<i;j++){
        w_time[i]+=btime[j];
   }
   total+=w_time[i];
```

```
    }

    avg_wait=(float)total/n;
    total=0;
    printf("\nProcess\t\tBurst time\t\twaiting time\t\tTurnaround Time");
    for(i=0;i<n;i++)
    {
       tat_t[i]=btime[i]+w_time[i];
       total=total + tat_t[i];
       printf("\nP%d\t\t\t%d\t\t\t%d\t\t\t%d",Proc_no[i],btime[i],w_time[i],tat_t[i]);
    }
    avg_tat=(float)total/n;
    printf("\n\nAverage waiting time = %f",avg_wait);
    printf("\n Average turnaround time = %f\n",avg_tat);

}
```