**Program 1:**

#include<iostream>

#include<conio.h>

using namespace std;

struct Student

{

bool resource\_pen;

bool resource\_paper;

bool resource\_question\_paper;

bool resource\_all\_resources;

}s1,s2,s3;

void student\_process\_one()

{

s1.resource\_all\_resources=1;

s1.resource\_paper=1;

s1.resource\_question\_paper=1;

cout<<"Student Process One Completed"<<endl;

}

void student\_process\_two()

{

s2.resource\_all\_resources=1;

s2.resource\_pen=1;

s2.resource\_question\_paper=1;

cout<<"Student Process Two Completed"<<endl;

}

void student\_process\_three()

{

s3.resource\_all\_resources=1;

s3.resource\_pen=1;

s3.resource\_paper=1;

cout<<"Student Process Three Completed"<<endl;

}

int main()

{

s1.resource\_all\_resources=0;s1.resource\_paper=0;s1.resource\_pen=0;s1.resource\_question\_paper=0;

s2.resource\_all\_resources=0;s2.resource\_paper=0;s2.resource\_pen=0;s2.resource\_question\_paper=0;

s3.resource\_all\_resources=0;s3.resource\_paper=0;s3.resource\_pen=0;s3.resource\_question\_paper=0;

do

{

int a,b;

cout<<"1.Pen"<<endl;

cout<<"2.Paper"<<endl;

cout<<"3.Question Paper"<<endl;

cout<<"Select any two items to be placed on the shared table:"<<endl;

cin>>a>>b;

if(a==1 && b==2 && s3.resource\_all\_resources==0)

{

student\_process\_three();

}

if(a==2 && b==1 && s3.resource\_all\_resources==0)

{

student\_process\_three();

}

if(a==2 && b==3 && s1.resource\_all\_resources==0)

{

student\_process\_one();

}

if(a==3 && b==2 && s1.resource\_all\_resources==0)

{

student\_process\_one();

}

if(a==1 && b==3 && s2.resource\_all\_resources==0)

{

student\_process\_two();

}

if(a==3 && b==1 && s2.resource\_all\_resources==0)

{

student\_process\_two();

}

}

while(s1.resource\_all\_resources==0||s2.resource\_all\_resources==0||s3.resource\_all\_resources==0);

cout<<"All Student Processes Completed";

getch();

}

**Program 2:**

#include<stdio.h>

int main()

{

int arrival\_T[10],burst\_T[10],rt[10],endTime,i,smallest;

int remain=0,n,j,avgwait=0,avgturnaround=0;

printf("Enter no of Processes : ");

scanf("%d",&n);

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

{

printf("Enter arrival time for Process P%d : ",i+1);

scanf("%d",&arrival\_T[i]);

printf("Enter burst time for Process P%d : ",i+1);

scanf("%d",&burst\_T[i]);

rt[i]=burst\_T[i];

}

printf("\n\nProcess\t|Turnaround Time| Waiting Time\n\n");

rt[9]=9999;

for(j=0;remain!=n;j++)

{

smallest=9;

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

{

if(arrival\_T[i]<=j && rt[i]<rt[smallest] && rt[i]>0)

{

smallest=i;

}

}

rt[smallest]--;

if(rt[smallest]==0)

{

remain++;

endTime=j+1;

printf("\nP[%d]\t|\t%d\t|\t%d",smallest+1,endTime-arrival\_T[smallest],endTime-burst\_T[smallest]-arrival\_T[smallest]);

avgwait+=endTime-burst\_T[smallest]-arrival\_T[smallest];

avgturnaround+=endTime-arrival\_T[smallest];

}

}

printf("\n\nAverage waiting time = %f\n",avgwait\*1.0/n);

printf("Average Turnaround time = %f",avgturnaround\*1.0/5);

return 0;

}