**OPERATING SYSTEMS (UCS303)**

***ASSIGNMENT NO. 4***

**2CS5 GROUP 3**

HIMANSHU NAGPAL – 102017110

KHUSHI GOEL – 102017112

MITALI JAIN – 102017113

DATE: AUGUST 30, 2021

**CPU SCHEDULING PROGRAMS**

**Q1. FCFS CPU Scheduling Program.**

|  |  |  |
| --- | --- | --- |
| **Process** | **Arrival Time** | **CPU Burst** |
| P1 | 0 | 5 |
| P2 | 1 | 1 |
| P3 | 2 | 2 |

**Ans: Code:**

#include <iostream>

using namespace std;

void waitingTime(int bt[], int wt[], int at[], int n){

int i, st[n];

st[0]=at[0];

wt[0]=0;

//service time

for(i=1; i<n; i++){

st[i]=st[i-1]+bt[i-1];

//waiting time for current process

wt[i]=st[i]-at[i];

if(wt[i]<0) wt[i]=0;

}

}

void turnAroundTime(int bt[], int wt[], int at[], int tat[], int n){

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

tat[i]=bt[i]+wt[i];

}

int main(){

int n=3;

int bt[n]={5,1,2};

int at[n]={0,1,2};

int wt[n], tat[n];

float tot\_wt=0, tot\_tat=0;

waitingTime(bt, wt, at, n);

turnAroundTime(bt, wt, at, tat, n);

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

cout<<endl<<"Process P"<<i+1<<endl;

cout<<"Waiting Time: "<<wt[i]<<endl;

cout<<"Turn Around Time: "<<tat[i]<<endl;

tot\_wt+=wt[i]; tot\_tat+=tat[i];

}

cout<<"\nAverage Waiting Time: "<<tot\_wt/n<<endl;

cout<<"Average Turn Around Time: "<<tot\_tat/n<<endl;

return 0;

}

Text

Description automatically generated

**Q2. SJF (Non-Pre-emptive) CPU Scheduling Program.**

|  |  |  |
| --- | --- | --- |
| **Process** | **Arrival Time** | **CPU Burst** |
| P1 | 0 | 5 |
| P2 | 1 | 1 |
| P3 | 2 | 2 |

**Ans: Code:**

#include <bits/stdc++.h>

using namespace std;

void swap(int x, int y){

int temp=x;

x=y;

y=x;

}

void waitingTime(int wt[], int bt[], int at[], int n){

wt[0]=0;

int sum=0;

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

sum+=bt[i-1];

wt[i]=sum-at[i];

}

}

void turnaroundTime(int tat[], int bt[], int wt[], int n){

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

tat[i]= bt[i]+wt[i];

}

int main(){

int i,j,k=1,n=3;

int p[]={1,2,3};

int bt[]={5,1,2};

int at[]={0,1,2};

int wt[n],tat[n];

double wtsum=0,tatsum=0;

//sort by arrival time:

for(i=0; i<n-1; i++)

{

for(j=i+1; j<n; j++)

{

if(bt[i]>bt[j])

{

swap(at[i],at[j]);

swap(bt[i],bt[j]);

swap(p[i], p[j]);

}

}

}

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

int b=0;

b+=bt[i];

int min=bt[k];

for(j=k; j<n; j++){

if(b>at[i] && bt[i]<min){

swap(p[j],p[k]);

swap(at[j],at[k]);

swap(bt[j],bt[k]);

}

}

k++;

}

waitingTime(wt,bt,at,n);

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

wtsum+=wt[i];

turnaroundTime(tat,bt,wt,n);

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

tatsum+=tat[i];

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

cout<<endl<<i+1<<". Process p"<<p[i]<<endl;

cout<<"waiting time "<<wt[i];

cout<<"\nturn around time "<<tat[i]<<endl;

}

cout<<"\nAverage waiting time "<<wtsum/n;

cout<<"\nAverage turn around time "<<tatsum/n;

return 0;

}

Text

Description automatically generated

**Q3. SRJF (Pre-emptive) CPU Scheduling Program.**

|  |  |  |
| --- | --- | --- |
| **Process** | **Arrival Time** | **CPU Burst** |
| P1 | 0 | 5 |
| P2 | 1 | 1 |
| P3 | 2 | 2 |

**Ans: Code:**

#include <bits/stdc++.h>

using namespace std;

void waiting\_time(int wt[], int bt[], int at[], int n){

int rt[n];

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

rt[i]=bt[i];

int p=0, t=0, flag=0, x=0, m=INT\_MAX, finish\_time;

// Process until all processes gets completed

while(p!=n){

// Find process with minimum remaining time among the

//processes that arrives till the current time

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

if(at[i]<=t && rt[i]<m && rt[i]>0){

m=rt[i];

x=i;

flag=1;

}

}

if(flag==0){

t++; continue;

}

//reduce remaining time, update minumum

rt[x]--;

m=rt[x];

if(m==0) m=INT\_MAX;

if(rt[x]==0){

p++; flag=0;

finish\_time=t+1;

//waiting time:

wt[x]=finish\_time-bt[x]-at[x];

if(wt[x]<0) wt[x]=0;

}

t++;

}

}

void turn\_around\_time(int wt[], int bt[], int tat[], int n){

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

tat[i] = bt[i] + wt[i];

}

int main(){

int i, n=3;

int bt[n] = {5, 1, 2};

int at[n] = {0, 1, 2};

int wt[n], tat[n];

float total\_wt=0, total\_tat=0;

waiting\_time(wt, bt, at, n);

turn\_around\_time(wt, bt, tat, n);

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

cout<<"\nProcess "<<i+1<<endl;

cout<<"Waiting time: "<<wt[i]<<endl;

cout<<"Turn around time: "<<tat[i]<<endl;

total\_wt+=wt[i];

total\_tat+=tat[i];

}

cout<<"\nAverage Waiting Time: "<<total\_wt/n<<endl;

cout<<"Average Turn Around Time: "<<total\_tat/n<<endl;

return 0;

}

Text

Description automatically generated