

Report on created Code

Assignment

**Supervisors:**

**Dr. Khaled Mahar**

**T.A: Sohaila**

Subject: System Moodelling

Name: Mazen Hazem

ID:18104084

**Abstract:**

My model is Designed to make it able to take interarrival and Service Time inputs from the user in any of the two different ways ( a)using given value b)mapping their values according to random number generated) through if condition statement otherwise will print error as a result the program will be able to calculate these attributes interarrival time, arrival time, Service time, service begin, Waiting time in queue, system time, idle time, number of customers in queue **and first service begin value** will be displayed as an output in the form of displayed array of the calculated results.

Then, this results will be calculated from the values resulted from the previous attributes:

1) average waiting time

2) probability of waiting

3) probability of server idling

4) average service time

5) average inter-arrival time (time between arrivals)

6) average waiting time (for those that do wait)

7) average time in the system

8) server utilization

9) average queue length

and finally you will be able to see these results’ values [and my attributes calculated values after just inputing their size of your inputs interarrival time and arrival time inputs.] and then input your given interarrival time and arrival time inputs sequentially required for the calculation.

**The Equations that i have derived to find my my attributes’ values that need to be calculated in my code are here in this algorithm:**

**ALGORITHM:**

Service **End** time=current arrival+service ( but if arrival<old end [End time=old end+current service])

Service **Begin** time=current arrival (if current arrival>old end **or** if current arrival = 0) **but**

=old End(if current arrival < old end)

**Waiting time** = old end – current arrival (but if result is <= 0 (-ve) therefore Waiting time=0)

**System time**=Cuurent End-arrival;

**Idle time**=the current arrival time-old end time (if result is negative answer is zero)

**Customers in queue**=count++(if current arrival < previous Service ends **do count++ for each previous value less than your cuurent arrival time value).**

**Second type of input in the else condition:**

is by inputing interarrival and Service time category number ,probability, then calculating the cummulative probability and generating random numbers that will then be mapped to it’s nearest category which will then be the values of the interarrival and service times and finally the program will be able to do the same functions and output the right results.

**The other required results needed [for my attributes outputs] will be calculated ass follows:**

n.b: Sum is the summation of values of your firstly inputed size for your [interarrival and service time inputs] while size is the size of your inputed values:

**Algorithm:**

1) average waiting time = sumofwaiting/size

2) probability of waiting = number(entities\_waited)/size

3) probability of server idling = sumofidling time/ sumoftimeinsystem

4) average service time = sumofservice time/size

5) average inter-arrival time (time between arrivals) = sumofinterarrival time/size

6) average waiting time (for those that do wait) = sumofwaiting time/size

7) average time in the system = sumoftimeinsystem/size

8) server utilization = 1-(sumofidletime/last end time)

9) average queue length = sumofcustomersinqueue/size

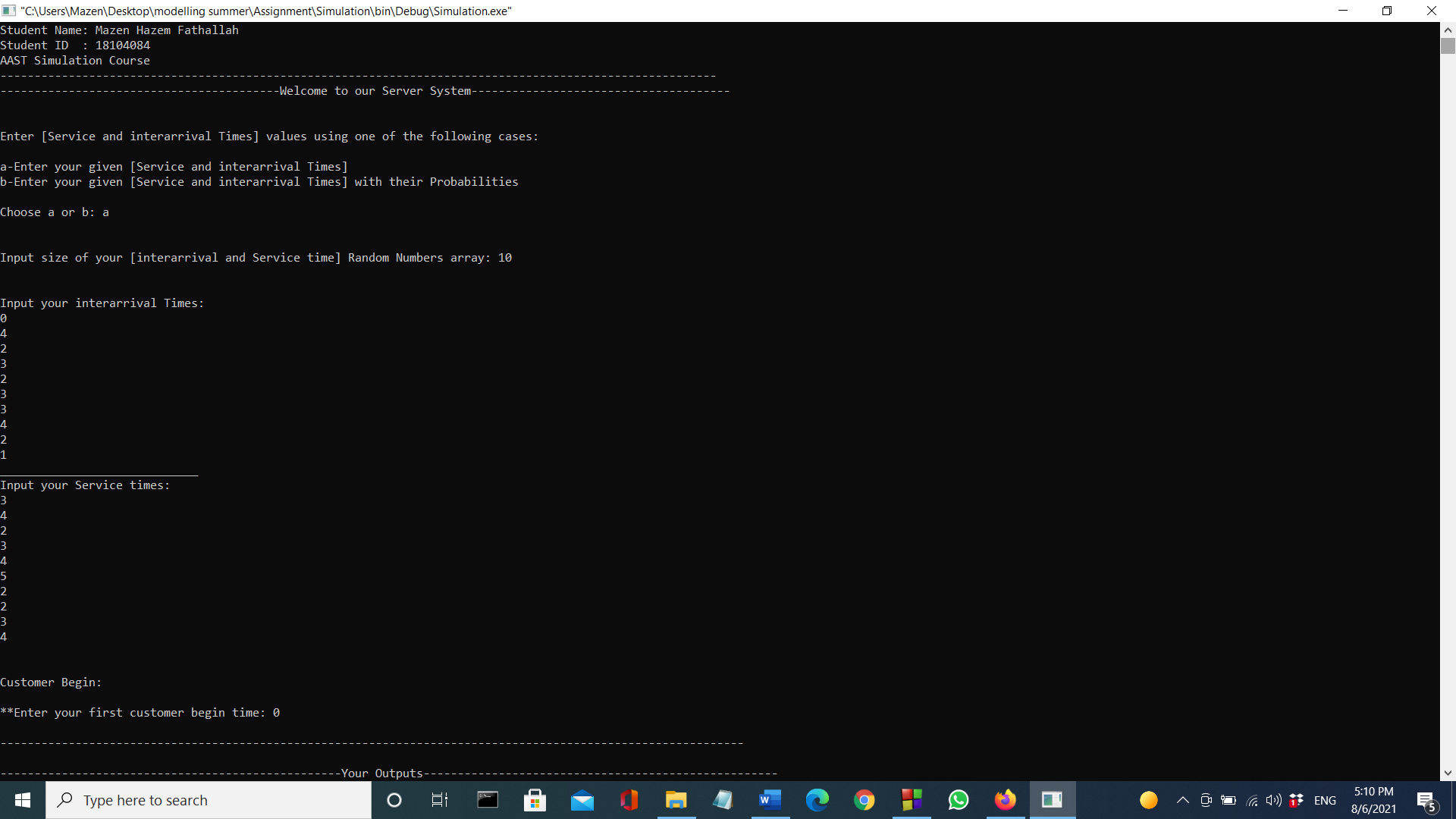
**Programming Language:**

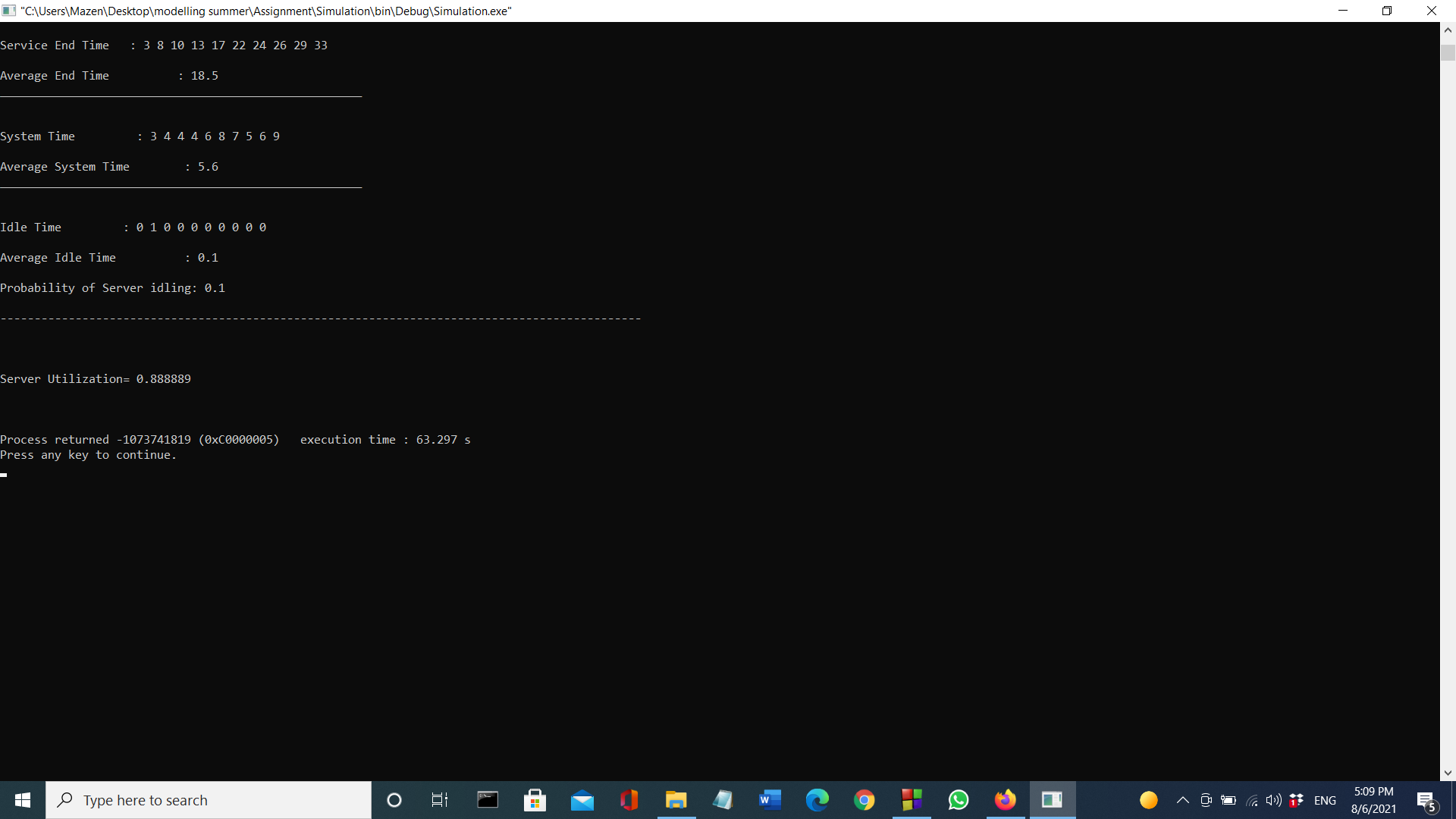
**C++ Language is the Programming language used to create this model.**

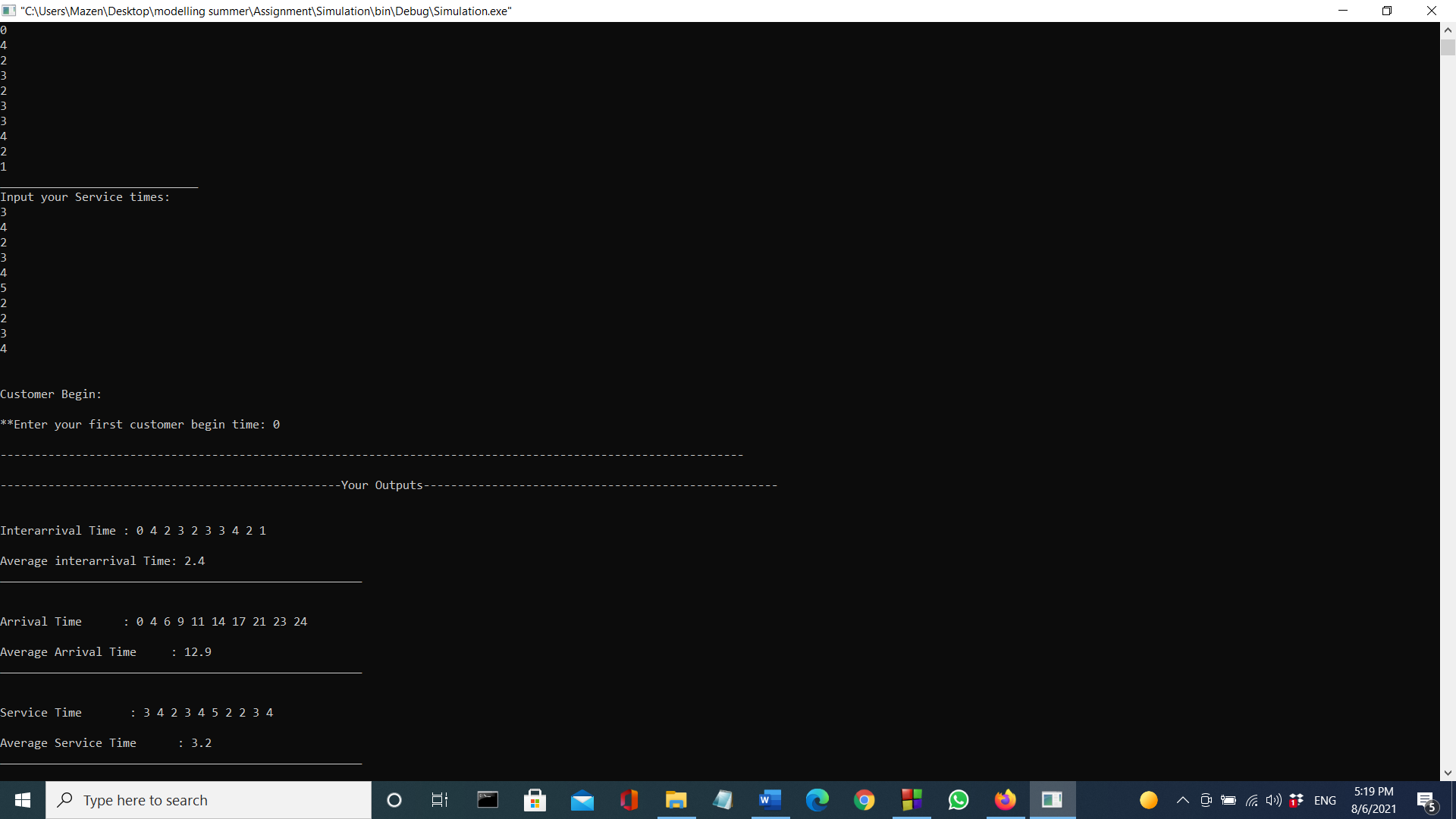
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

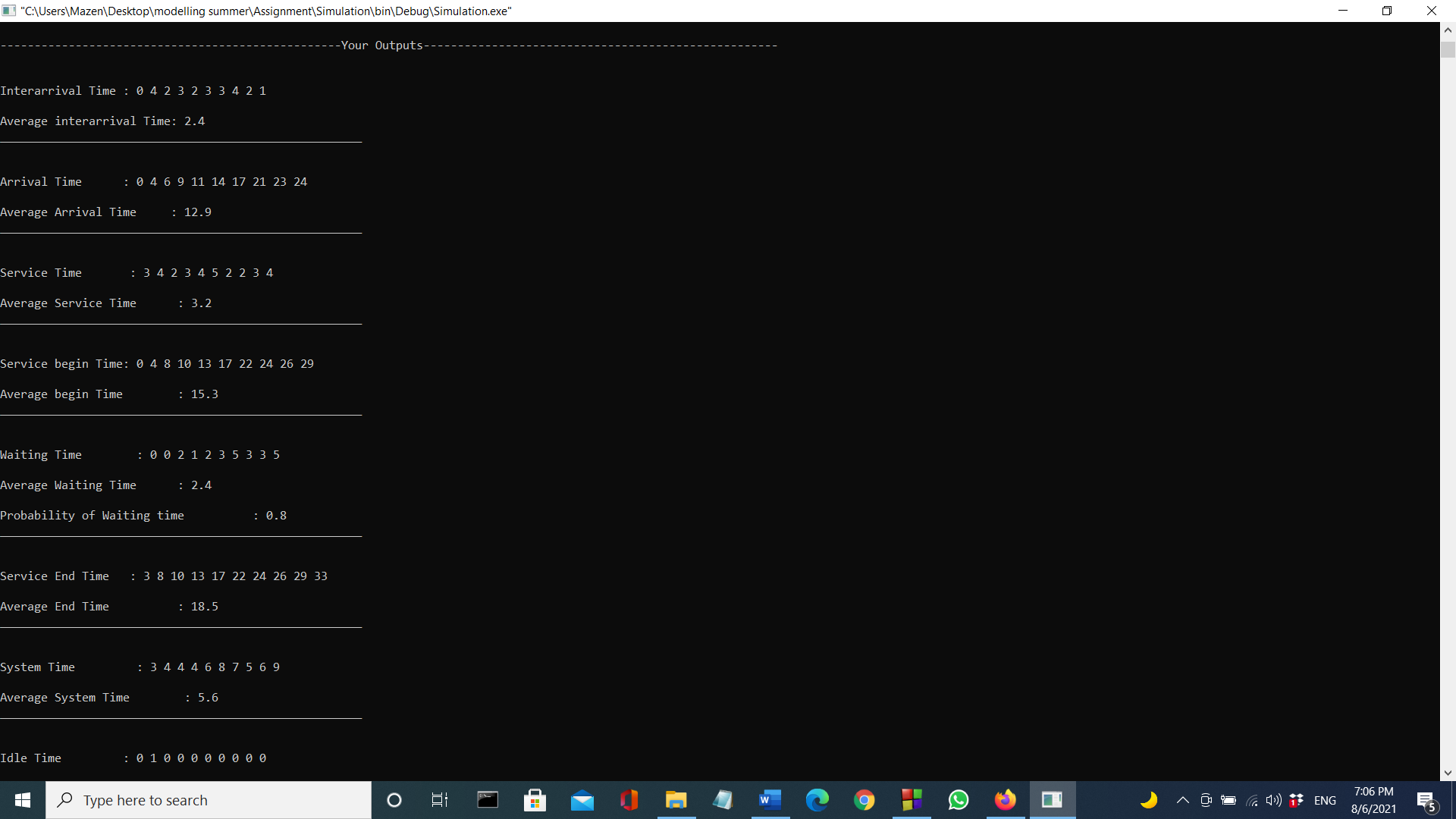
**Observation:**

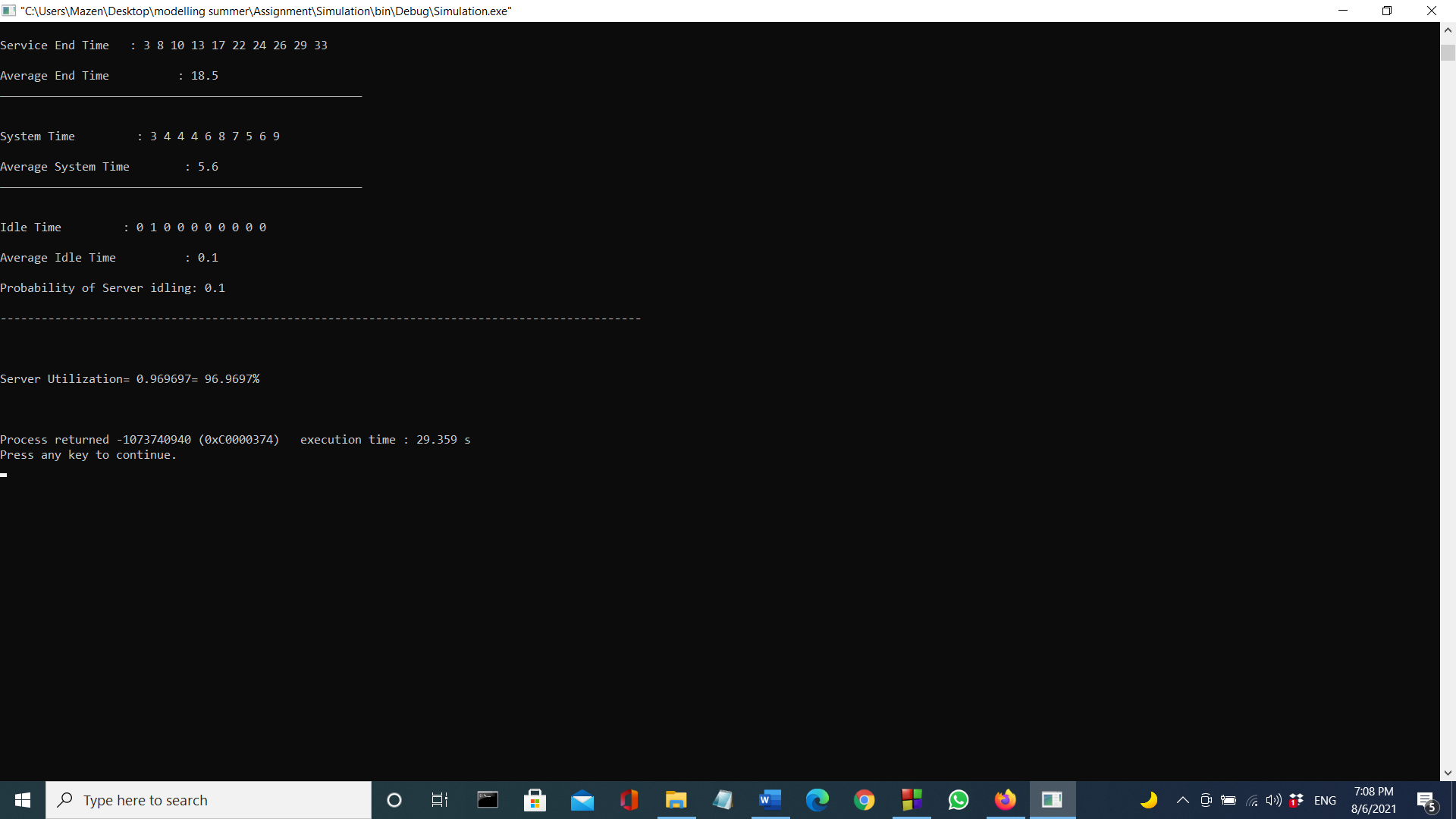
**ScreenShoots:**

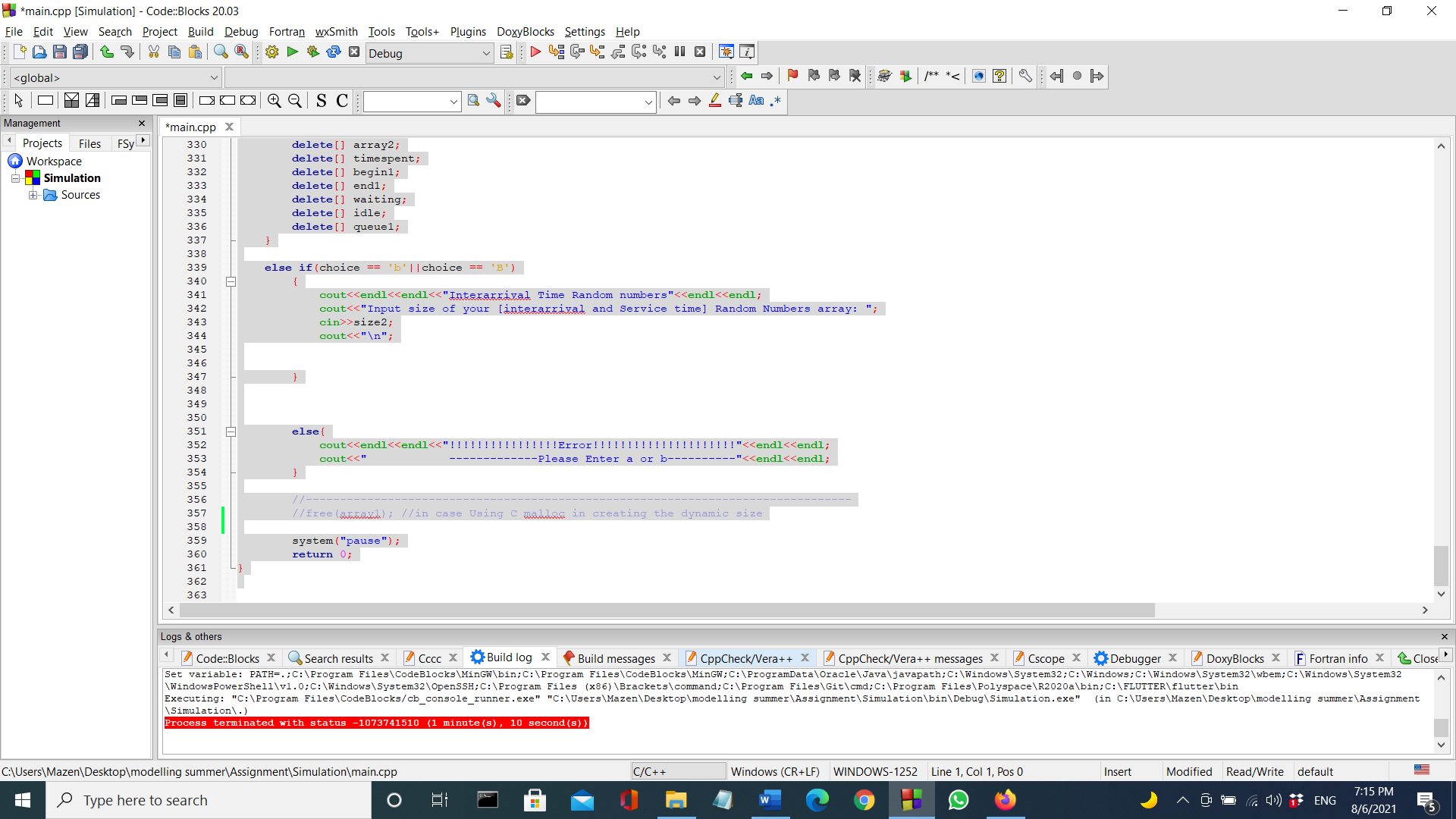












**Code:**

#include <iostream>

#include "time.h"

#include <math.h>

#include <cstdlib>

#include <iomanip> //for having exact decimal points in line 119

#include<random>

using namespace std;

int main()

{

cout<<"Student Name: Mazen Hazem Fathallah"<<endl;

cout<<"Student ID : 18104084"<<endl;

cout<<"AAST Simulation Course"<<endl;

cout<<"---------------------------------------------------------------------------------------------------------"<<endl;

cout<<"-----------------------------------------Welcome to our Server System--------------------------------------"<<endl<<endl<<endl;

//First Step Create a dynamic array to input your Givens

std::random\_device rd;

std::mt19937 gen(rd());

char choice;

int size2;

int size3;

//int \*probarray=new int[size2];

//int mod1=100;

//int mod1total=0;

//float x=(float) rand()/RAND\_MAX; //used in line 112

//float y; //used in line 147

srand(time(0)); //m3 kol run m4 hytla3ly nafs al Random numbers sawa2 fy al service b2a aw al interarrival aw 2ay random numbers h3mlaha input

cout<<"Enter [Service and interarrival Times] values using one of the following cases:"<<endl<<endl;

cout<<"a-Enter your given [Service and interarrival Times]"<<endl;

cout<<"b-Enter your given [Service and interarrival Times] with their Probabilities"<<endl<<endl;

cout<<"Choose a or b: ";

cin>>choice;

if (choice == 'a'||choice == 'A')

{

cout<<endl<<endl<<"Input Size of Your Schedule: ";

cin>>size2;

int \*array1=new int[size2];

//int size3;

int \*array2=new int[size2];

int \*arrival=new int[size2];

int \*begin1=new int[size2];

int \*end1=new int[size2];

int \*waiting=new int[size2];

int \*idle=new int[size2];

int \*timespent=new int[size2];

int \*queue1 = new int[size2];

//cout<<endl<<endl<<"Interarrival Time Random numbers"<<endl<<endl;

//cout<<"\n";

//Using C

//int \*array1=(int\*)malloc(sizeof(int)\*size2);

//Using C++ //line 16

cout<<endl<<endl<<"Input your interarrival Times: "<<endl;

for(int i=0;i<size2;i++)

{

cin>>array1[i];

}

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<"Input your Service times: "<<endl;

for(int i=0;i<size2;i++)

{

cin>>array2[i];

}

arrival[0]=array1[0];

for(int i=0;i<size2;i++)

{

//cout<<arrival[i]<<"\t";

arrival[i+1]=arrival[i]+array1[i+1];

}

//cout<<endl<<endl<<"Average Arrival Time :"<<sum2/size2;

double sum3=0;

for(int i=0;i<size2;i++)

{

sum3+=array2[i];

}

double sum5=0;

for(int i=0;i<size2;i++)

{

end1[i]=arrival[i]+array2[i];

if(i==0)

{

end1[i]=arrival[i]+array2[i];

//cout<<end1[i]<<" ";

}

else

{

if(arrival[i]>=end1[i-1]){

end1[i]=arrival[i]+array2[i];

//cout<<end1[i]<<" ";

}

else if(arrival[i]<end1[i-1])

{

end1[i]=end1[i-1]+array2[i];

//cout<<end1[i]<<" ";

}

}

sum5+=end1[i];

}

//cout<<endl<<"Your average service end array : "<<sum5/size2;

double sum4=0;

cout<<endl<<endl<<"Customer Begin: ";

for(int i=0;i<size2;i++)

{

//begin1[i]=arrival[i];

if(i==0)

{

cout<<endl<<endl<<"\*\*Enter your first customer begin time: ";

cin>>begin1[i];

if(arrival[i]==begin1[i])

{

begin1[i]=arrival[i];

//cout<<begin1[i]<<" ";

}

}

else

{

if(arrival[i]<end1[i-1])

{

begin1[i]=end1[i-1];

//cout<<begin1[i]<<" ";

}

else if(arrival[i]>=end1[i-1])

{

begin1[i]=arrival[i];

//cout<<begin1[i]<<" ";

}

}

sum4+=begin1[i];

}

//cout<<endl<<"Your average service begin array : "<<sum4/size2;

//cout<<endl<<endl<<"Your waiting time array : ";

//double sum8=0;

double count1;

double sum8=0;

for(int i=0;i<size2;i++)

{

waiting[i]=begin1[i]-arrival[i];

if(waiting[i]!=0)

{

count1++;

}

sum8+=waiting[i];

}

//cout<<endl<<"Your average waiting time array : "<<sum8/size2;

/\*

double sum9=0;

int k=0;

for(int i=0;i<size2;i++)

{

if(i==0)

{

queue1[i]=0;

}

else{

for(int j=i;j>=0;j--){

if(arrival[j]<end1[i-1])

{

queue1[i]=k+1;

}

else

{

queue1[i]=0;

}

}

}

sum9+=queue1[i];

\*/

//cout<<endl<<endl<<"Your time spent in system array : ";

double sum6=0;

for(int i=0;i<size2;i++)

{

timespent[i]=end1[i]-arrival[i];

//cout<<timespent[i]<<"\t";

sum6+=timespent[i];

}

//cout<<endl<<"Your average time spent array : "<<sum6/size2;

//cout<<endl<<endl<<"Your idle time in system array : ";

double sum7=0;

for(int i=0;i<size2;i++)

{

idle[i+1]=arrival[i+1]-end1[i];

if(arrival[i]-end1[i-1]<=0)

{

idle[i]=0;

//cout<<idle[i]<<"\t";

}

sum7+=idle[i];

}

cout<<endl<<"-------------------------------------------------------------------------------------------------------------"<<endl;

cout<<endl<<"--------------------------------------------------Your Outputs----------------------------------------------------"<<endl;

double sum=0;

cout<<endl<<endl<<"Interarrival Time : ";

for(int i=0;i<size2;i++)

{

cout<<array1[i]<<" ";

sum+=array1[i];

}

cout<<endl<<endl<<"Average interarrival Time: "<<sum/size2;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

double sum2=0;

cout<<endl<<endl<<"Arrival Time : ";

for(int i=0;i<size2;i++)

{

cout<<arrival[i]<<" ";

sum2+=arrival[i];

}

double s2=sum2/size2;

cout<<endl<<endl<<"Average Arrival Time : "<<s2;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Service Time : ";

for(int i=0;i<size2;i++)

{

cout<<array2[i]<<" ";

}

double s3= sum3/size2;

cout<<endl<<endl<<"Average Service Time : "<<s3;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Service begin Time: ";

for(int i=0;i<size2;i++)

{

cout<<begin1[i]<<" ";

}

double s4= sum4/size2;

cout<<endl<<endl<<"Average begin Time : "<<s4;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Waiting Time : ";

for(int i=0;i<size2;i++)

{

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

}

cout<<endl<<endl<<"Average Waiting Time : "<<sum8/size2;

cout<<endl<<endl<<"Probability of Waiting time : "<<count1/size2;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

/\*

cout<<endl<<endl<<"Customers in Queue : ";

for(int i=0;i<size2;i++)

{

cout<<queue1[i]<<" ";

}

//cout<<endl<<endl<<"Average Customers in queue : "<<sum9/size2;

\*/

//cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Service End Time : ";

for(int i=0;i<size2;i++)

{

cout<<end1[i]<<" ";

}

cout<<endl<<endl<<"Average End Time : "<<sum5/size2;;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"System Time : ";

for(int i=0;i<size2;i++)

{

cout<<timespent[i]<<" ";

}

cout<<endl<<endl<<"Average System Time : "<<sum6/size2;

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Idle Time : ";

for(int i=0;i<size2;i++)

{

cout<<idle[i]<<" ";

}

cout<<endl<<endl<<"Average Idle Time : "<<sum7/size2;

cout<<endl<<endl<<"Probability of Server idling: "<<sum7/sum6;

cout<<"\n\n"<<"----------------------------------------------------------------------------------------------"<<endl;

cout<<endl<<endl<<endl<<"Server Utilization= "<<(1-(sum7/end1[size2-1]))<<"= "<<(1-(sum7/end1[size2-1]))\*100<<"%"<<endl;

cout<<"\n\n";

delete[] array1;

delete[] array2;

delete[] arrival;

delete[] array2;

delete[] timespent;

delete[] begin1;

delete[] end1;

delete[] waiting;

delete[] idle;

delete[] queue1;

}

else if(choice == 'b'||choice == 'B')

{

cout<<endl<<endl<<"Interarrival Time Random numbers"<<endl<<endl;

cout<<"Input size of your [interarrival and Service time] Schedules array : ";

cin>>size2;

double \*random1=new double[size2];

double \*interprob=new double[size2];

double \*intercategories=new double[size2];

double \*interprobcummulative=new double[size2];

double \*random2=new double[size2];

double \*serviceprob=new double[size2];

double \*servicecategory=new double[size2];

double \*servicecummulative=new double[size2];

cout<<endl<<"Input size of your [interarrival and Service time] Random Numbers array : ";

cin>>size3;

//cout<<"\n";

cout<<endl<<"Enter your interarrival Categories : "<<endl;

for(int i=0;i<size2;i++)

{

cin>>intercategories[i];

}

cout<<"Enter your interarrival Probabilities : "<<endl;

for(int i=0;i<size2;i++)

{

cin>>interprob[i];

}

interprobcummulative[0]=interprob[0];

cout<<"Your cummulative interarrivalprobability : ";

for(int i=0;i<size2;i++)

{

//cout<<arrival[i]<<"\t";

interprobcummulative[i+1]=interprobcummulative[i]+interprob[i+1];

cout<<interprobcummulative[i]<<" ";

}

cout<<endl<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Random numbers to calculate interarrival Time: ";

//srand(time(0));

for(int i=0;i<size3;i++)

{

random1[i]=rand()%100;

//random1[i]=random1[i]/100;

cout<<random1[i]<<" ";

}

cout<<endl<<endl;

/\*

for(int i=0;i<size3;i++)

{

for(int j=i;j<size2;j++){

if(random1[i]<interprobcummulative[j])

{

random1[i]=intercategories[j];

//cout<<random1[i]<<" ";

}

else//if(random1[i]>=interprobcummulative[j])

{

random1[i]=intercategories[j+1];

//cout<<random1[i]<<" ";

}

}

}

\*/

cout<<endl<<"--------------------------------------------------------------------------------------------------------------"<<endl;

cout<<"\n\n";

cout<<"Service Time Random numbers"<<endl<<endl;

cout<<"Enter your Service Time Categories : "<<endl;

for(int i=0;i<size2;i++)

{

cin>>servicecategory[i];

}

cout<<"Enter your Service time Probabilities : "<<endl;

for(int i=0;i<size2;i++)

{

cin>>serviceprob[i];

}

cout<<"your cummulative service time probabilities: ";

servicecummulative[0]=serviceprob[0];

for(int i=0;i<size2;i++)

{

//cout<<arrival[i]<<"\t";

servicecummulative[i+1]=servicecummulative[i]+serviceprob[i+1];

cout<<servicecummulative[i]<<" ";

}

cout<<"\n"<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<endl<<endl<<"Random numbers to calculate Sevice Time: ";

//srand(time(0));

for(int i=0;i<size3;i++)

{

//random2[i]=random2[i]/100;

random2[i]=(rand()%100);

cout<<random2[i]<<" ";

}

cout<<endl<<endl;

for(int i=0;i<size3;i++)

{

for(int j=i;j<size2;j++){

if(random2[i]<servicecummulative[j])

{

random2[i]=servicecategory[j];

cout<<random2[i]<<" ";

}

else if(random2[i]>=servicecummulative[j])

{

random2[i]=servicecategory[j+1];

cout<<random2[i]<<" ";

}

}

}

cout<<endl<<endl;

delete[] intercategories;

delete[] interprob;

delete[] interprobcummulative;

delete[] servicecategory;

delete[] serviceprob;

delete[] servicecummulative;

delete[] random1;

delete[] random2;

}

else{

cout<<endl<<endl<<"!!!!!!!!!!!!!!!!Error!!!!!!!!!!!!!!!!!!!!!"<<endl<<endl;

cout<<" -------------Please Enter a or b----------"<<endl<<endl;

}

//--------------------------------------------------------------------------------

//free(array1); //in case Using C malloc in creating the dynamic size

system("pause");

return 0;

}