**Computer Simulation-Group project: ICS 4D**

Group Members:

1. Peris Nyambura
2. Claire Wambui
3. Faith Njuguna
4. Cecilia Agonga
5. Nicole Angwenyi

**Assignment:**

Assume a uniform arrival of customers at the bank and that the random arrival times are uniformly distributed on(1,8) minutes and that the service times are randomly distributed on (1,6) minutes. Create a program in Java/C++ to simulate 100 customers and hence evaluate queue statistics in a table as shown in the example above

**C++ code implementation:**

#include <iostream>

#include <iomanip>

#include <vector>

#include <random>

#include <algorithm>

struct Customer {

int id;

double arrivalTime;

double serviceTime;

double serviceStart;

double serviceEnd;

double waitingTime;

double timeInSystem;

};

int main() {

const int NUM\_CUSTOMERS = 100;

std::vector<Customer> customers;

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_real\_distribution<> arrivalDist(1.0, 8.0); // Uniform(1, 8)

std::uniform\_real\_distribution<> serviceDist(1.0, 6.0); // Uniform(1, 6)

double lastServiceEnd = 0.0;

std::cout << std::fixed << std::setprecision(2);

std::cout << std::setw(10) << "Customer"

<< std::setw(15) << "Arrival Time"

<< std::setw(15) << "Service Time"

<< std::setw(15) << "Service Start"

<< std::setw(15) << "Waiting Time"

<< std::setw(15) << "Service End"

<< std::setw(18) << "Time in System" << "\n";

for (int i = 1; i <= NUM\_CUSTOMERS; ++i) {

Customer c;

c.id = i;

c.arrivalTime = arrivalDist(gen);

c.serviceTime = serviceDist(gen);

c.serviceStart = std::max(c.arrivalTime, lastServiceEnd);

c.waitingTime = c.serviceStart - c.arrivalTime;

c.serviceEnd = c.serviceStart + c.serviceTime;

c.timeInSystem = c.serviceEnd - c.arrivalTime;

customers.push\_back(c);

lastServiceEnd = c.serviceEnd;

std::cout << std::setw(10) << c.id

<< std::setw(15) << c.arrivalTime

<< std::setw(15) << c.serviceTime

<< std::setw(15) << c.serviceStart

<< std::setw(15) << c.waitingTime

<< std::setw(15) << c.serviceEnd

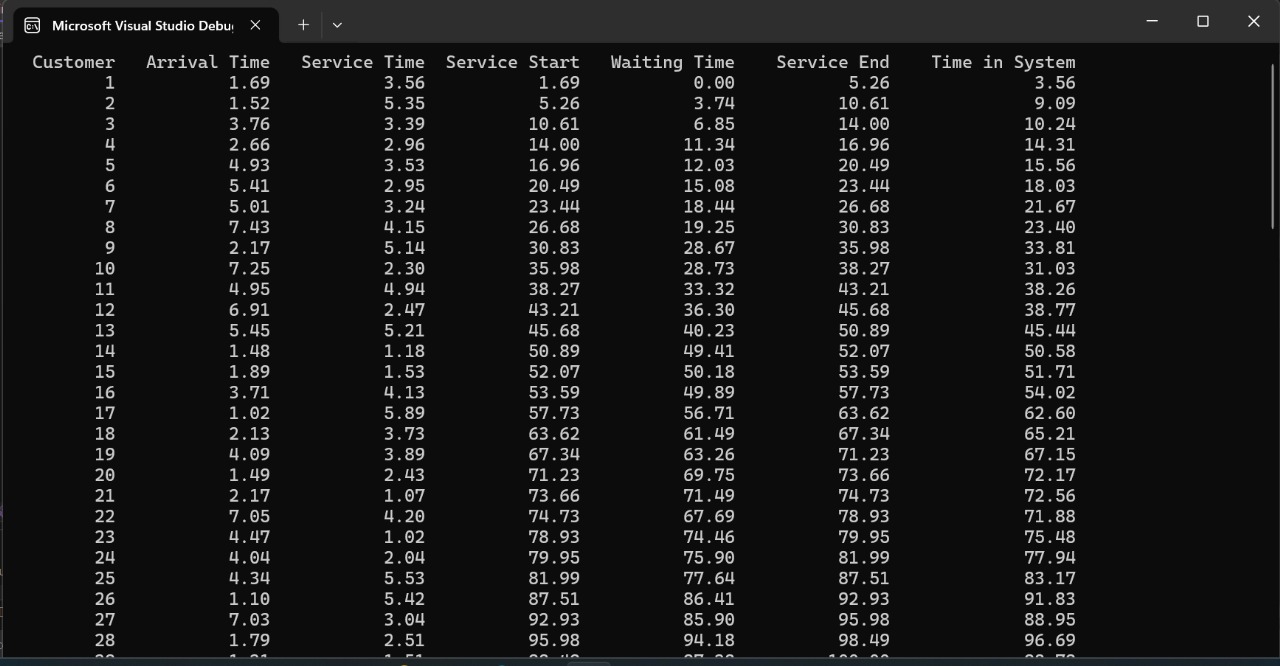
<< std::setw(18) << c.timeInSystem << "\n";

}

return 0;

}

**Output:**



A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.