Trent University
COIS 2020

Test Case for Assignment 2

Date: November 12 2023

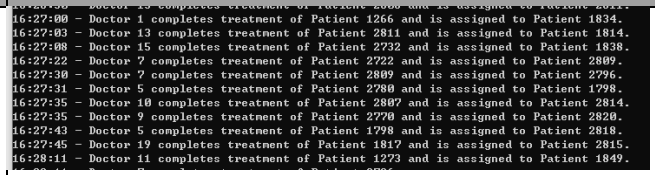
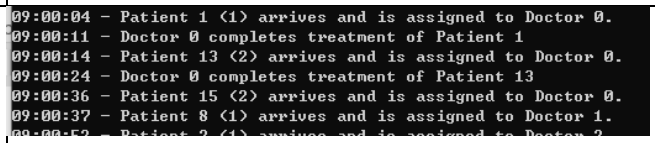
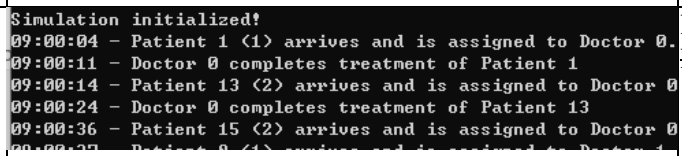
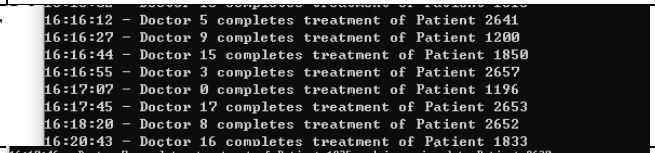
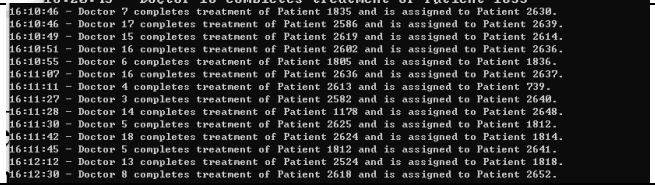
Version 1.0

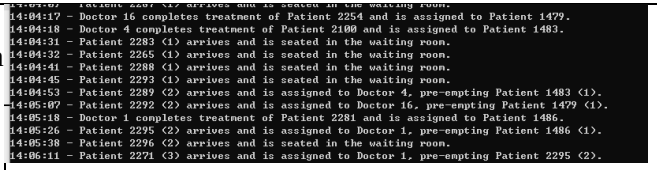
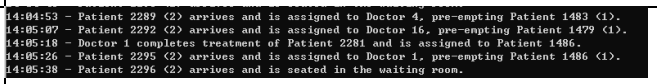
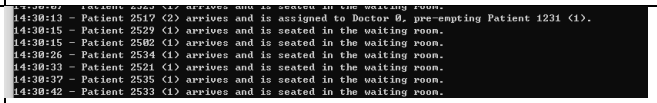
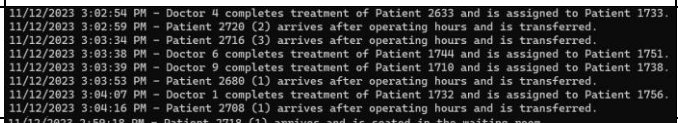
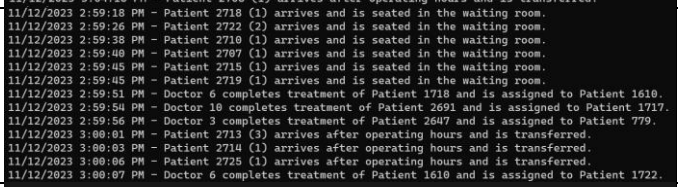
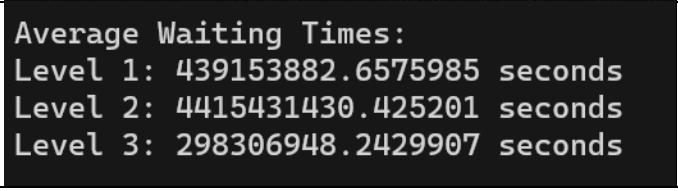
Teammates

David Chan, 0767384
Mohammad Abdur Rakib, 0685509
Chengjun Yin, 0695866

1. Purpose

The purpose of the document is to provide the test plan after completing Patient, Event, PriorityQueue, and Simulation classes for a discrete event simulation simulating hospital emergency rooms.

Test Case #	Task	Test to be performed	Expected Result	Actual result	Pass/ Fail
1	Patient Class- Inter-Arrival Time Generation Test	Run the random number generator part of the program to verify the average generated time is close to M. Input: M = 60 (1-minute mean inter-arrival time)	1000 randomly generated inter-arrival times with an average close to M.	 More than 1000 of patients are generated	Pass/ ~~Fail~~
2	Patient Class- Treatment Time Generation Test for each Emergency Level	Run the random treatment time generator to check if the means for each level are close to T, 2T, and 4T, respectively. Input: T = 120 (10 minutes mean treatment time)	Random treatment times for each emergency level (1 = T, 2 = 2T, 3 = 4T) with average times close to the respective values.		Pass/ ~~Fail~~
3	Patient Class- Patient Arrival Event Handling	The ARRIVAL events are generated and placed in the Priority Queue correctly. Input: Simulate patient arrivals at 9:00	Patients are correctly instantiated and queued for treatment		Pass/ ~~Fail~~
4	Event Class- Patient Departure Event Handling	The DEPARTURE events are generated when treatment is concluded and that they're handled properly. Input: Simulate patient departure at 12:00	Patients leave the system after their treatment		Pass/ ~~Fail~~
5	Event Class- Doctor Assignment to New Patient Upon Arrival	A patient arrives when a doctor is available. Verify that the ARRIVAL event results in immediate doctor assignment.	Patient is immediately assigned to a doctor		Pass/ ~~Fail~~

6	Doctor Assignment to New Patient With Waiting Queue	A patient arrives when no doctor is available. Check if the patient is correctly queued in the absence of an available doctor.	Patient is placed into the appropriate waiting queue based on their emergency level		Pass/ ~~Fail~~
7	Pre-emption Logic Check	Verify that the pre-emption logic correctly adjusts the Priority Queue and assigns the higher priority patient to a doctor. Input: A higher emergency level patient arrives when all doctors are busy with lower emergency level patients	Lower emergency level patient is pre-empted, and the higher emergency patient is given immediate treatment		Pass/ ~~Fail~~
8	Treatment Time Adjustment Post Pre-emption	Check if the pre-empted patient's DEPARTURE event is accurately modified to reflect the new remaining treatment time. Input: Pre-emption of an ongoing treatment event	The remaining treatment time of the pre-empted patient is updated correctly		Pass/ ~~Fail~~
9	Patients Arriving at Closing Time	Confirm that the cut off time is inclusive for patient arrivals. Input: Patients arriving at exactly 15:00	They should be seen by a doctor, not turned away		Pass/ ~~Fail~~
10	Patients Arriving After Closing Time	Make sure no ARRIVAL events are processed post-closing and the patients are handled accordingly. A patient arrives post 15:00	The patient should be turned away (simulated transfer to another hospital)		Pass/ ~~Fail~~
11	Calculating Average Waiting Times for Patients	Make sure to print the average waiting times for patients according to their emergency level.	The program should print 3 separate sections (emergency level) for average waiting times.		Pass/ ~~Fail~~