Home assignment 2, Simulation 2014

Send the home assignment to Saeed Bastani in a pdf file. Also put the code in the pdf file.

In task 1 it the process scheduling approach should be used for task 1. Remember that it is always good to answer the following questions before you start coding:

- What process classes are needed?
- What variables are needed in the process classes?
- What shall a process do when it receives a signal?

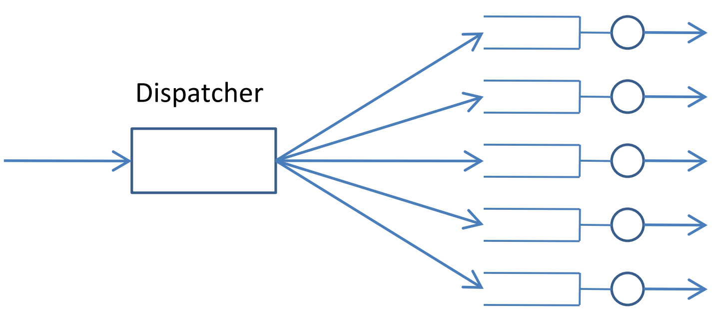
As usual, spend some time on answering these questions before you start coding; it often saves a lot of time later on.

Task 2 is an example of a problem where you must simulate a course of events several times, in this case a pharmacy.

Task 3, finally, is an example of simulating the lifetime of a system that consists of several components. In this case you have to draw all random numbers (the life-time of the components) at the beginning of the simulation. Also in this case you have to simulate a course of events several times.

Task 1

You shall study load sharing among a number of service stations. Examples of this can be found in web servers, communication networks and logistic systems. Each service station has a server and a buffer. All items that arrive to the system first come to a dispatcher. The dispatcher chooses to which service station an item shall be sent. See the figure:



There are 5 queuing systems. The service times exponentially distributed with mean 0.5 second in all the queuing systems. The dispatcher can chose to which queue it shall send a new job to in one of the following ways:

i. Random, which means that with equal probability one of the queuing systems is chosen.

ii. Round robin, which means that the first customer is sent to queuing system 1, the second to 2… the fifth to 5 and after that the next one is sent to 1 etc.
iii. We send the job to the queuing system with the smallest number of jobs, if there is more than one system with the smallest number of jobs one of them is chosen randomly with equal probability.

The time between the arrivals to the dispatcher have a uniform distribution. The time used by the dispatcher to choose between the queuing systems is very small so we assume that it is 0.

You shall do the following:

1. Write a simulation program for the system above. All three algorithms for load distribution above shall be implemented.
2. Verify your program using little's theorem for all three algorithms above. Do that for a mean arrival time of 0.12 seconds.
3. Find the mean number of jobs in the queuing systems for all the algorithms and the following mean arrival times to the dispatcher: 0.11, 0.15 and 2.00 seconds. Which is the best algorithm?

Simulate the system for at least 100 000 seconds.

Task 2

Consider a pharmacist who is thinking of setting up a pharmacy where he will fill prescriptions. He plans to open at 9.00 every weekday. No new prescriptions are accepted after 17.00. During the opening hours customers come according to a Poisson process with rate 4 per hour. The time it takes to fill a prescription is uniformly distributed between 10 and 20 minutes. The prescriptions are always handled in a FIFO manner. After 17.00 he has to continue working until all prescriptions that have arrived during the day are filled in. Use simulation to answer the following questions:

a) What is the average time when his work will have finished every day?
b) What is the average time from the arrival of a prescription until it has been filled in?

Observe that you have to simulate many working days to be able to answer the questions! Simulate at least 1000 working days.

Task 3

Assume that we have a system that consists of five components; each one of them has a uniformly distributed life length in the interval from 1 to 5. We also assume that if component 1 breaks down, also component 2 and 5 breaks down and if component 3 breaks down also component 4 breaks down. There are no other dependencies between the life lengths of the components. The system works as long as at least one component works. Find the mean time until the system breaks down. Simulate at least 1000 "runs" of the system.