# COIS 4470H Assignment 3

## Question 1: GPSS 01

Language: GPSS

**Interarrival Time** = RVEXPO(1,30) #in seconds

**Service Time** = 40\_-4 seconds

1. <=0.58 go to Morning Paper Guy. Everyone else goes to Wall Street Journal Guy (even who are buying both papers)

Text

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Vendor1 = Morning Paper Guy

Vendor2 = Wall Street Journal Guy

|  |  |  |
| --- | --- | --- |
| **Vendor** | **Number of People/ Contents** | **Time** |
| **Morning Paper Guy** | 124 | Total: 0.683  Average: 39.642 |
| **Wall Street Journal** | 91 | Total: 0.498  Average: 39.435 |

|  |  |  |
| --- | --- | --- |
| **Queue** | **Number of People/ Contents** | **Time** |
| **Line1 (Morning)** | Total: 127  Max: 4  Average: 0.660 | Average Time: 37.441 |
| **Line2 (Wall Street)** | Total: 93  Max: 4  Average: 0.219 | Average Time: 16.972 |

1. **How to make the system more efficient?**

As we can see, the average time in the Queue for the Morning Paper Vendor, is very high as compared to Wall Street Journal Guy. Since the probability of people buying morning paper is 58, which is almost 60% - that is a high probability.

We know that Wall Street Journal Vendor has both Morning Paper and Wall Street Journals. Therefore, To optimize the system, we can implement the method where the incoming customers for morning papers are DISTRIBUTED among the two vendors. That is, the customer who wants to buy the morning paper will be sent to the Wall Street Journal Vendor if the Wall Street Journal has the shorter queue

Text

Description automatically generated

A picture containing text, black, screen, close

Description automatically generated

As we see, the average time for the Morning Paper line decreased drastically.

One more thing that we could do, the very obvious thing is add another vendor. This way we have 3 servers now so that would decrease the total time. But having one more vendor depends on ‘budget’ realistically speaking. So, the above solution would be better option to make it efficient.

## Question 2: GPSS 02

|  |  |  |  |
| --- | --- | --- | --- |
| **Performance Measure** | **LoadSplit** | **TurnTaker** | **ShortQ** |
| Server Utilization – Server 1 | Total = 0.770  Avg time/Xact = 100.000 | Total = 0.822  Avg time/Xact = 100.00 | Total = 0.934  Avg time/Xact = 98.765 |
| Server Utilization – Server 2 | Total = 0.910  Avg time/Xact = 98.801 | Total = 0.859  Avg time/Xact = 98.472 | Total = 0.857  Avg time/Xact = 100.000 |
|  |  |  |  |
| Number of request served - Server 1 | 46 | 49 | 52 |
| Number of request served - Server 2 | 54 | 51 | 48 |
|  |  |  |  |
| Average number in queue – queue for server 1 | 2.010 | 2.482 | 0.990 |
| Average number in queue – queue for server 2 | 1.936 | 2.843 | 0.613 |
|  |  |  |  |
| Mean delay time - queue for server 1 | 239.981 | 269.082 | 102.747 |
| Mean delay time - queue for server 2 | 195.947 | 325.884 | 71.509 |
|  |  |  |  |
| Mean Response Time – Server 1 (total waiting time for each request) | 339.981 | 369.082 | 201.512 |
| Mean Response Time – Server 2 | 294.748 | 424.356 | 171.509 |

The obvious recommendation would be the “ShortQ”. The server utilization depends on the shorter queue length, which means, there is not going to be an unnecessary crowd and load on one server just because of some ‘random number probability’.

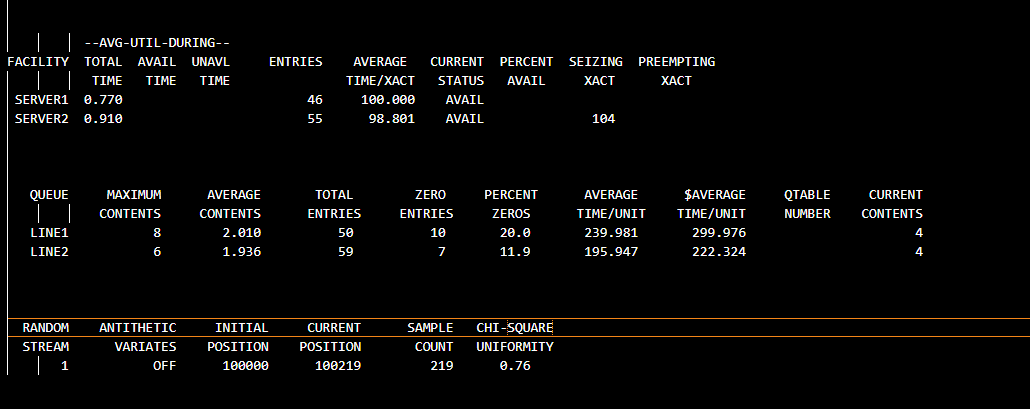
Taking the statistics and performance of the simulation into consideration to support my recommendation –

* The server utilization for both the servers are ‘equal’ or similar. This means, that both servers are being used in a similar way and just one server does NOT have excess load
* The number of requests served by both the Web servers are again, similar – (51 and 49). Since we have assumed that all the requests take exactly 100.00 time for processing therefore, then this similarity indicates that both servers’ utilization is very close
* Average number in queue and the mean delay time as you can see makes a BIG difference. The requests are being serviced at a much faster rate as compared to other servers
* Overall, The mean response time is the lowest for ShortQ and hence the best approach

**Load Split**

Graphical user interface

Description automatically generated with medium confidence



**Turn Taker**

Text

Description automatically generated

A picture containing graphical user interface

Description automatically generated

**ShortQ**

A picture containing graphical user interface

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

## Question 3: Random Number Generator

Lehmer random-number generator

**xi+1 = g(x) = axi mod 251** *i=0,1,2,3…….*

m = 251

1. Using Theorem: “If m is prime and p1, p2, . . . , pr are the (unique) prime factors of m − 1”

m-1 = 250 = 2 \* 53

number of full-period multipliers = (p1-1)(p2-1)(m-1)/(p1\*p2)

= (2-1)(5-1)(251-1)/(2\*5)

= (250\*4)/10 = 100

There are **100** full – period multipliers

1. **Program:**



Shape

Description automatically generated

Please check entire output in attached file A3\_all\_multipliers.txt

1. **Output File: A3\_random\_numbers.txt**

Generating random numbers in range (0,1)

x0 = 3

Text

Description automatically generated

1. **Output File: A3\_100RN**

Making a change to code so 100 random variates are generated

Text

Description automatically generated

Text

Description automatically generated

1. **Chi squared test**

Chi squared statistic value generated = 12.2

For alpha = 0.01 and s = 10, (degree of freedom = 9) chi critical value = 21.7

12.2 is less than 21.7, therefore, TRUE all values generated are uniform

I anyways implemented the calculation of this in the program itself. As you see, it says “True” for being uniformly generated

Text

Description automatically generated

1. **GAP Test (0.2, 0.5)**

**Output File: A3\_GAP\_Out.txt**

Random numbers are independent : )

Text

Description automatically generated