CSCE 2211

Applied Data Structures

Prof. Amr Goneid, AUC

Simulation of a Waiting Line

- Queues can simulate waiting lines
- A waiting line is a queue of jobs waiting to be served by a server on FCFS (First Come First Serve) basis.



Prof. Amr Goneid, AUC

Time of arrival of a job is essentially random, but with a fixed probability per unit time.

- A clock registers arrival time, the simulation is Time-Driven
- Simulation tries to answer the question: What happens if...

Prof. Amr Goneid, AUC

Notations:

- T_{max} Maximum Simulation Time (fixed)
- *t* Clock Time = current time ($0 \le t < T_{max}$)
- ΔT Average time between two arrivals (fixed)
- $P_{arrival}$ Probability of arrival per unit time = 1/ ΔT
- T_s Service time (fixed)
- T_{rem} Time remaining to start service
- T_{arrival} Arrival time (random)

 T_{wait} Wait time = $t - T_{arrival}$

Prof. Amr Goneid, AUC



Prof. Amr Goneid, AUC

How to simulate a clock and what happens during a clock tick

Set T_{max} , Initialize clock time t = 0while $(t < T_{max})$

Arrival Processing

Service Processing

Prof. Amr Goneid, AUC

<u>Algorithm</u>

Set T_{max} , $T_{arrival}$, T_s and compute $P_{arrival} = 1 / \Delta T$ Initialize waiting line and clock t = 0set $T_{rem} = 0$ while $(t < T_{max})$ { 1. Test for arrival. If job arrived, process arrival. 2. Test for server ready. If ready, exit line, start service and compute wait time 3. If $(T_{rem} > 0)$ decrement T_{rem} 4. Increment clock time t *Compute average wait time*

Test for Arrival & Arrival Processing to queue Q

Arrival (Q, t)

 $\begin{array}{ll} Generate \ Random \ number \ (R) \ between \ 0 \ and \ 1.0 \\ if \ (R < P_{arrival}) & // \ job \ arrived \\ if \ (Q.queueIsFull()) \ report \ error: \ Line \ is \ full \\ else \ \{T_{arrival} = t \ ; \ Q.enqueue \ (T_{arrival}); \ \} \end{array}$

<u>Test for server ready. If ready, exit line and start</u> <u>service.</u>

Service (Q, t, T_{rem}, waitTotal, jobcount)
{
 if((T_{rem} is 0) and (not Q.queueIsEmpty()))
 {
 ExitLine (Q, t, waitTotal, jobcount)
 set T_{rem} = T_s; // start service
 }
}

Prof. Amr Goneid, AUC

Exit Line

10

```
ExitLine (Q, t, waitTotal, jobcount)
{
    if (Q.queueIsEmpty()) Report: Line is Empty
        else
        {
            Q.dequeue (t<sub>a</sub>);
            T<sub>wait</sub> = t - T<sub>arrival</sub>; // wait time
            waitTotal = waitTotal + T<sub>wait</sub>; // Total wait time
            jobcount = jobcount + 1; // jobs serviced
        }
}
```

Compute average wait time

Averagewait (waitTotal, jobcount)

if (jobcount is 0) return 0; else return (waitTotal / jobcount);

Arrival# 5 at: 31

Job# 4 Service Started at: 33 wait = 9 Job# 5 Service Started at: 39 wait = 8 Arrival# 6 at: 46 Job# 6 Service Started at: 46 wait = 0 Arrival# 7 at: 48 Arrival# 8 at: 50 Arrival# 9 at: 52 Job# 7 Service Started at: 52 wait = 4 Job# 8 Service Started at: 58 wait = 8 Arrival# 10 at: 64 Job# 9 Service Started at: 64 wait = 12 Arrival# 28 at: 278

Job# 28 Service Started at: 278 wait = 0

Average wait Time is 2.89286

A possible tracing of the simulation program

12