**Example 8.7**

Suppose the call centeris currently staffed by only one CSR who takes an average of *Tp* =2.5 minutes to process a call, so her processing rate is 1/*Tp =* 0.4 customers per minute or 24 per hour. With number of servers *c* =1, the processing capacity is also *Rp* = 24 per hour. Suppose customer calls come in at an average in 3 minute intervals, so the average arrival rate is *Ri =* 1/3 per minute or20 per hour. Since *Rp > Ri*, the process is stable. However, variability in arrivals and processing will result in delays and queues. We will assume the interarrival and processing times are exponentially distributed so we can use the spreadsheet **Performance.xls** to calculate process performance characteristics.

Suppose L. L. Bean estimates that the retailer loses $2 in sales for every minute that a customer has to wait on line for a CSR, in terms of dissatisfaction with service as well as the resulting impact on future sales to disgruntled customers. Since *Ii* is the average number of callers waiting in line, the average total cost of waiting will be $2 *Ii* per minute or $120 *Ii* per hour. Alternately, each caller waits an average of *Ti* minutes for a CSR at a cost of $2 *Ti*, while an average of *Ri* customers call in every minute, so the total cost of waiting is $2 *RiTi* per minute or $120 *RiTi* per hour. However, by Little’s law, we know *Ri* *Ti* = *Ii*, so the waiting cost is again $2 *Ii* per minute or $120 *Ii* per hour. We can use **Performance.xls** spreadsheetto calculate *Ii* for different number of servers *c* = 1, 2, . . . . Obviously, as we hire more servers, the cost of waiting will go down, while the cost of providing service will go up. Suppose each CSRis paid $20 an hour, so with *c* servers, the hourly cost of providing service will be $20*c.*  The manager of the L. L. Bean call centerwould like to determine the optimal number of CSRs to minimize

Total hourly cost = $20*c* + $120 *Ii*.

We compute and tabulate this cost by using **Performance.xls** with following inputs:

*c =* 1, 2, . . . .

*Ri* = 20/hour

1/*Tp* = 24/hour

And obtain hourly costs as summarized in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *c* | *Ii* | $20*c* | $120 *Ii* | Total Hourly Cost |
| 1 | 4.167 | $20 | $500.04 | $520.04 |
| 2 | 0.175 | $40 | $21.00 | $61.00 |
| 3 | 0.022 | $60 | $2.64 | $62.64 |
| 4 | 0.003 | $80 | $0.36 | $80.36 |

Thus, the total hourly cost of waiting and providing service is minimized when the number of CSRs is *c* = 2.

Alternately, L. L. Bean may be concerned with the *total* turnaround time *T* that a customer spends for the entire transaction, including waiting for a CSR *and* being served. In that case, L. L. Bean*’*s problem is to determine *c* that minimizes

Total hourly cost = $20*c* + $120*I*.

The results are summarized below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *c* | *I* | 20*c* | 120 *I* | Total Cost |
| 1 | 5.000 | $20 | $600.00 | $620.00 |
| 2 | 1.008 | $40 | $120.96 | $160.96 |
| 3 | 0.856 | $60 | $102.72 | $162.72 |
| 4 | 0.836 | $80 | $100.32 | $180.32 |

Again, *c* = 2 minimizes the total hourly cost of providing service and customer’s total time spent in the system.