**Konrad Bartlett**

**0580964**

**COIS 4470**

**Assignment 2**

1. **Inventory system: An automobile dealership uses a weekly periodic inventory review policy. Assume the maximum space for cars is S=80 and the minimum inventory level is s=20. Operation costs are assumed as:**
2. Modify the program sis1.c or the sis1.java to compute all four components of the total average cost per week. (use the input file sis.dat)

**Holding cost (C\_holding) - $25 per car per week Line 87**

**while**(input.hasNextInt()) {

...

c.**holding** += inventory \* 25;

}

**Shortage cost (C\_Shortage) - $700 per car per week Line 112**

System.out.print(Math.*round*(sum.**shortage** \* 700));

**Set up cost (C\_SetUp) - $1000 per order Line 61**

**if**(inventory < MIMIMUM)

c.**setup** += 1000;

**Unit cost (C\_Unit) - $8000 each car ordered Line 114**

System.out.print(Math.*round*(sum.**order** \* 8000));

1. Use your program to compute and complete the following table (S=80):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***s*** | ***0*** | ***5*** | ***10*** | ***15*** | ***20*** | ***25*** | ***30*** | ***35*** | ***40*** |
| ***Average holding*** | **$460.00** | **$538.00** | **$538.00** | **$577.25** | **$687.75** | **$777.75** | **$841.75** | **$896.00** | **$911.25** |
| ***Average shortage*** | **$795.77** | **$374.34** | **$374.34** | **$345.50** | **$172.47** | **$15.88** | **$1.48** | **$0.28** | **$1.19** |
| ***Average setup*** | **$310.00** | **$330.00** | **$330.00** | **$340.00** | **$380.00** | **$430.00** | **$460.00** | **$490.00** | **$500.00** |
| ***Sum*** | **$1,565.77** | **$1,242.34** | **$1,242.34** | **$1,262.75** | **$1,240.22** | **1,223.63** | **$1,273.23** | **$1.386.28** | **$1,412.44** |

1. What could be the optimum value for s? Explain.

The optimum value for s is approximately 25. This is because at this value we see this is the first value where almost no cars put on order (which costs $1000 each time). This greatly reduces the sum cost, despite also slightly increasing setup and holding cost.

1. Re-do (a)-(c) with the following assumptions that backorder is not permitted and set up cost depends on the number of items ordered (no change on other cost):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***s*** | ***0*** | ***5*** | ***10*** | ***15*** | ***20*** | ***25*** | ***30*** | ***35*** | ***40*** |
| ***Average holding*** | **$460.00** | **$538.00** | **$538.00** | **$577.25** | **$687.75** | **$777.75** | **$841.75** | **$896.00** | **$911.25** |
| ***Average shortage*** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** |
| ***Average setup*** | **$372.00** | **$396.00** | **$396.00** | **$400.00** | **$424.00** | **$462.00** | **$480.00** | **$496.00** | **$504.00** |
| ***Sum*** | **$832.00** | **$934.00** | **$934.00** | **$977.25** | **$1,111.75** | **$1,239.70** | **$1,621.75** | **$1,392.00** | **$1,415.25** |

**The optimum value for *s* is 0. This is because the car dealer no longer accepts orders (shortages) for cars which always resulted in costing them a lot of money. Because no more cost is associated with ordering vehicles the only attribute that affects price becomes how many vehicles are there, (holding cost).**

1. Modify your program in (d) to output the number of customers who were not satisfied (left without getting a car) every week.

Line 80

**if**(inventory > demand )   
**else**  
 c.**demand** += demand - inventory;

1. Re-do (a)-(c) with the assumption that customers who were not satisfies will come again in the following week (additional demand for the following week).

Line 70

c.**shortage** = c.**demand**;  
demand = input.nextInt() + c.**shortage;**

**if**(inventory > demand )   
 **else**c.**order** = demand - inventory;

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***s*** | ***0*** | ***5*** | ***10*** | ***15*** | ***20*** | ***25*** | ***30*** | ***35*** | ***40*** |
| ***Average holding*** | **$19.50** | **$112.25** | **$323.00** | **$298.75** | **$602.75** | **$734.25** | **$790.75** | **$886.50** | **$825.25** |
| ***Average shortage*** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** | **$0.00** |
| ***Average setup*** | **$552.00** | **$564.00** | **$552.00** | **$556.00** | **$510.00** | **$498.00** | **$510.00** | **$496.00** | **$520** |
| ***Sum*** | **$571.50** | **$676.25** | **$875.00** | **$854.75** | **$1,112.75** | **$1,232.25** | **$1,300.75** | **$1,382.50** | **$1,345.25** |

The optimum value for **s** is 0 again for the same reason. This dealer is limited to how many cars they are provided not by how many customers arrive. They are provided with the same number of cars, so the simulation is not much different than question D) where we were also limited by the number of cars available to the dealer.

1. **Develop a Monte Carlo simulation program to calculate an approximation π by considering the number of random points selected inside the quarter circle: : 1, 0, 0 2 2 Q x + y = x ≥ y ≥ , where the quarter circle is taken to be inside the square: S : 0 ≤ x ≤ 1and 0 ≤ y ≤ 1. Use the equation: π/4 = (area Q) / (area S).**
2. Using your program to fill in the following form:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***# points*** | 102 | 103 | 104 | 105 | 106 |
| ***π*** | 3.16 | 3.164 | 3.1376 | 3.1462 | 3.140404 |

1. **Three dice are rolled and the largest of the three up faces is recorded. Let X be this value, the possible values of X are 1, 2, …, 6. Use Monte Carlo simulation to estimate the probability of each possible value. Do the experiment for 1000 times and record the results in the following table. The largest number X 1 2 3 4 5 6 Estimated Probability**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***Largest side*** | 1 | 2 | 3 | 4 | 5 | 6 |
| ***Probability*** | 0.3% | 2.7% | 9.4% | 17% | 28.8% | 41.8% |

1. **GPSS Simulation: Suppose cars arrive at the drive-through lane of a fast-food restaurant**

* **with interarrival times of 10±10 minutes. It takes 3±1 minutes for the driver of a car to place an order over the intercom (the intercom is a single server). The car then spends 15±3 seconds moving from the intercom to the pick up window (another single server). A car spends 3±2 minutes at the pick up window while the order is paid for, bagged, and handed over to the driver. The car then leaves.**

1. Model this system to gather Queue information that applies to cars from the time they arrive to beginning service at the intercom.
2. Extend the model of (a) to so that it will additionally gather Queue information that applies to cars from the time they arrive to having been serviced at the intercom.
3. Extend the model of (b) to so that it will additionally gather Queue information that applies to cars waiting their turn for service at the window.
4. How many cars have less then 20 minutes of total time in system (from arrival to departure)? How many cars have total time in system more than 20 but less then 25?

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Model A*** | ***Model B*** | ***Model C*** |
| ***Mean delay in queue*** |  |  |  |
| ***Mean number in queue*** |  |  |  |

|  |  |
| --- | --- |
| ***Total time in system (Model D)*** | ***Number of Cars*** |
| ***< 20*** |  |
| ***20 – 25*** |  |
| ***25 <*** |  |