OPERATION RESEARCH ASSIGNMENT GROUP 14

**Question:**

Consider the single period model with C 0 = Re 1; Cs = Rs 19 per unit and I = 1.12 units. The demand density function is given by f(D) = f(D) = De ^(-D). The cost of carrying inventory if left after the period is Re 1 per unit.

Simulate the system for 50 days and estimate the mean number of papers should he order each day.

**Assumptions:**

1. Reorder point of the inventory is 0 units i.e. a new order is placed when the inventory level reaches 0 units.
2. Lead time is 0 i.e. stock replenishment takes instantaneously as on soon as the order is placed.
3. Maximum inventory level is assumed to be 2 units. Replenishment of stock is accomplished using (0,2) policy i.e. whenever the inventory level reaches 0 units, a quantity is ordered so that the inventory reaches a maximum level of 2 units.
4. No back ordering is allowed.
5. Initial inventory is given to be 1.12 units.
6. Cost of ordering one unit is Re 1, Shortage Cost is Rs. 19 per unit and holding cost is Re 1 per unit.
7. Partial orders are fulfilled.

**Variables (along with their initialized values):**

* day=1;
* invb (initial inventory)=1.12;
* imax (maximum inventory)=2;
* dd (demand)=0;
* inve (Ending Inventory)=0;
* tor (Quantity to be ordered)=0;
* sht (shortage)=0;
* c0 (Ordering Cost)=1;
* c1 (Holding Cost)=1;
* cs (Shortage Cost)=19;
* rand1: Two random numbers are generated. They are multiplied and stored in the variable rand1.
* i,k,t,j,ra=Variables to run the loops.
* An array of 50 rows and 10 columns is created to store the data.

**Flowchart**

START

Initialize : It is assumed that initially 1.12 units are there in inventory.

day=1, invb=1.12,dd=0, inve=0, tor=0, str=0,c0=1, c1=1,c2=19,i=0

NO

i<50

YES

Generate Random Numbers for demand.

Generate Demand and store it in variable dd.

Compute:

Sht=dd-invb

Set inve= (-1)\*sht

Set tor=imax

Compute:

Inve=invb-dd

Set sht=0

Set tor=0

invb>=dd

NO YES

tor>0

NO YES

Meaning, No new stock is ordered.

Compute:

Total ordering cost= 0

Total Holding cost=c1\*inve

Total Shortage Cost=Cs\*sht

Stock is ordered.

Total ordering cost= C0\*tor

Total Holding cost=0

Total Shortage Cost=Cs\*sht

Compute invb=imax

Compute invb=inve

Compute:

Average number of orders= sum of orders/50

STOP

**CODE:**

**(To print in c code)**

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

#include<time.h>

#include<string.h>

int main()

{

int i=0,k,t,j,ra,day=0; //Variables are initialized

double rand1;

double invb,dd,inve,tor,imax,sht,c0,c1,cs,sum=0;

invb=1.12;

imax=2;

dd=0;

inve=0;

tor=0;

sht=0;

c0=1;

c1=1;

cs=19;

double arr[50][11];

//Random Numbers Generated to produce the demand Function

srand(time(NULL));

while (i<50)

{

ra=2;

j=2;

while(ra--)

{

arr[i][j] = (double)rand() / (double)RAND\_MAX;

j++;

}

rand1= arr[i][2]\*arr[i][3];

dd= (-1)\*log(rand1); //demand is generated;

if(invb>=dd) //checking if inventory in the beginning more than or equal to demand

{

inve=invb-dd;

sht=0;

tor=0;

}

else

{

sht=dd-invb;

inve=(-1)\*sht;

tor=imax;

}

if(tor>0) //costs are calculated on the basis if orders are made or not

{

c0=1;

c1=0;

cs=(-1)\*inve\*19;

}

else

{

c0=0;

c1=1\*inve;

cs=0;

}

// data stored in form of array

day=i+1;

arr[i][0]=day;

arr[i][1]=invb;

arr[i][4]=dd;

arr[i][5]=inve;

arr[i][6]=tor;

arr[i][7]=c0\*tor;

arr[i][8]=c1;

arr[i][9]=cs;

if(invb>=dd) // Inventory in beginning is set for next day on the basis of order received;

{

invb=inve;

}

else

{

invb=imax;

}

i++;

}

for(k=0;k<50;k++) // Output is generated

{

for(t=0;t<10;t++)

{

printf(" %lf ",arr[k][t]);

}

printf("\n");

}

for(j=0;j<50;j++)

{

sum=sum+arr[j][6];

}

printf("\n\nMean number of papers should he order each day: %lf units \n",sum/50);

}

**(To print in excel sheet)**

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

#include<time.h>

#include<string.h>

int main()

{

FILE \* fp = fopen("mydata1.csv", "w"); // opens file to write ("w"), using the FILE pointer "fp"

int i=0,k,t,j,ra,day=0; //Variables are initialized

double rand1;

double invb,dd,inve,tor,imax,sht,c0,c1,cs,sum=0;

invb=1.12;

imax=2;

dd=0;

inve=0;

tor=0;

sht=0;

c0=1;

c1=1;

cs=19;

double arr[50][11];

//Random Numbers Generated to produce the demand Function

srand(time(NULL));

while (i<50)

{

ra=2;

j=2;

while(ra--)

{

arr[i][j] = (double)rand() / (double)RAND\_MAX;

j++;

}

rand1= arr[i][2]\*arr[i][3];

dd= (-1)\*log(rand1); //demand is generated;

if(invb>=dd) //checking if inventory in the beginning more than or equal to demand

{

inve=invb-dd;

sht=0;

tor=0;

}

else

{

sht=dd-invb;

inve=(-1)\*sht;

tor=imax;

}

if(tor>0) //costs are calculated on the basis if orders are made or not

{

c0=1;

c1=0;

cs=(-1)\*inve\*19;

}

else

{

c0=0;

c1=1\*inve;

cs=0;

}

// data stored in form of array

day=i+1;

arr[i][0]=day;

arr[i][1]=invb;

arr[i][4]=dd;

arr[i][5]=inve;

arr[i][6]=tor;

arr[i][7]=c0\*tor;

arr[i][8]=c1;

arr[i][9]=cs;

if(invb>=dd) // Inventory in beginning is set for next day on the basis of order received;

{

invb=inve;

}

else

{

invb=imax;

}

i++;

}

fprintf(fp,"Day,Inventory in beginning,Random Number 1, Random Number 2, Demand,Ending Inventory,Quantity Ordered,Total Ordering Cost, Holding Cost,Shortage Cost\n");

for(k=0;k<50;k++) // Output is generated

{

for(t=0;t<10;t++)

{

fprintf(fp," %lf, ",arr[k][t]);

}

fprintf(fp,"\n");

}

for(j=0;j<50;j++)

{

sum=sum+arr[j][6];

}

fprintf(fp,"\n\nMean number of papers should he order each day: %lf units \n",sum/50);

fclose(fp); // close the file after you're done

}

**OUTPUT:**





