**SC & IS 545** Aks5800

Homework 1

1). Code for Simulation

Dim Clock As Double ' simulation clock

Dim NextFailure As Double ' time of next failure event

Dim NextRepair As Double ' time of next repair event

Dim S As Double ' system state

Dim Tlast As Double ' time of previous event

Dim Area As Double ' area under S curve

Dim Barea As Double ' area under utilization curve

Dim Busy As Double ' state of technician

Dim k As Double ‘Variable

Public Sub MainProgram()

' Program to generate a sample path for the reliability example

Dim NextEvent As String

k = 2

For NumReps = 1 To 100

S = k + 1

Clock = 0

Tlast = 0

Area = 0

Busy = 0

Barea = 0

NextFailure = WorksheetFunction.Floor(6 \* Rnd(), 1) + 1

NextRepair = 1000000

Do Until S = 0

NextEvent = Timer

Select Case NextEvent

Case "Failure"

Call Failure

Case "Repair"

Call Repair

End Select

Loop

Worksheets("Sheet2").Cells(NumReps, 1) = Clock

Worksheets("Sheet2").Cells(NumReps, 2) = Area / Clock

Worksheets("Sheet2").Cells(NumReps, 3) = Barea / Clock

Next NumReps

Average1 = Application.WorksheetFunction.Average(Sheet2.Range("a" & 1 & ":a" & 100))

Average2 = Application.WorksheetFunction.Average(Sheet2.Range("b" & 1 & ":b" & 100))

Average3 = Application.WorksheetFunction.Average(Sheet2.Range("c" & 1 & ":c" & 100))

MsgBox (" Average System failure in 100 replication " & Average1 & " with average # components " & Average2 & " and Utilization of Repair Technician= " & Average3)

End Sub

Public Function Timer() As String

' Determine the next event and advance time

If NextFailure < NextRepair Then

Timer = "Failure"

Clock = NextFailure

NextFailure = 1000000

Else

Timer = "Repair"

Clock = NextRepair

NextRepair = 1000000

End If

End Function

Public Sub Failure()

'Failure event

Area = Area + (Clock - Tlast) \* S

Barea = Barea + (Clock - Tlast) \* Busy

Tlast = Clock

S = S - 1

If S >= 1 Then ' there is a spare available to come on line

NextFailure = Clock + WorksheetFunction.Floor(6 \* Rnd(), 1) + 1

If S = k Then ' this is the only one under repair, so start repairing

NextRepair = Clock + 2.5

End If

End If

End Sub

Public Sub Repair()

'Repair event

Area = Area + (Clock - Tlast) \* S

Barea = Barea + (Clock - Tlast) \* Busy

Busy = 0 ' make temporarily idle

S = S + 1

If S < k + 1 Then ' there is still another to be repaired

NextRepair = Clock + 2.5

Busy = 1

End If

Tlast = Clock

End Sub

**b). Performance Measures**

Average system failure of 100 replication – 118.24

Average Component – 2.237

Utilization of Repair Technician – 0.198

**2). Description of the Code**

**Ans.**

a)**.** Modify the model so that there are k ≥ 1 spare components available instead of 1. That is, k should be a variable that can be set in the code. The components are still repaired one-at-a-time by a single repair technician.

Explanation: In the Code, k is set as a variable which can be inserted in the code. k are spared components and S equals k +1.

b). Collect statistics on the utilization of the repair technician. Hint: It will be helpful to have a variable, say Busy, that is 0 when the technician is idle, and 1 when busy. The utilization is the time-average value of this variable.

Explanation: In the Code, state of the repair technician is set as Busy with values 1 and 0. If technician is busy with repairing the component, it is set 1 and 0 for being idle in case every component is working fine. The time the technician works on repairing the component is recorded as Barea. This can be used to calculate utilization by dividing the recorded time to total time of the till system fails.

c). Set up the program to make 100 replications, printing out the three performance measures (TTF, average number of functional components and utilization) on each. Run the program with k = 2 and compute the across-replication average of each performance measure.

Explanation: The Code has been set with a ‘for’ loop to execute the simulation for 100 replication. K has been defined as number of spared components which be inserted into the program for different values. Do loop is been used to run the code till the system fail. Once system fails, the results are stored into the rows as 1- time till failure, 2- Average number of components, and third row as Utilization of repair technician.

Code consists of Main Program, and two sub program as failure and repair. Main program executes the main functions of for and do loops and give the results of performance measures. The sub programs are called from main function as next events changes with the timer. Timer is used through random generation of numbers and as event changes, timer records it.