

APPENDIX

COMPUTER PROGRAMMES

```
10 'estimation of parameters in M/G/1 with G = simple exponential'  
20 'and many server with c = 2 & c = 4 priority queueing models'  
30 DIM PAT(1000), NPAT(1000),PST(1000),NPST(1000)  
40 INPUT "enter total time of observation"; T  
50 'arrival time modules'  
60 GOSUB 1500  
70 'service time modules'  
80 GOSUB 1860  
90 'data file modules'  
100 GOSUB 2160  
110 GOSUB 2350  
120 'processing modules'  
130 PRINT "enter 1 for single server case"  
140 PRINT "enter 2 for many server case"  
150 INPUT RESPONSE  
160 IF RESPONSE = 2 THEN 630  
170 I = 1 : J = 1  
180 IF FAT(I) > NPAT(J) THEN 260  
190 IDT = PAT(I)  
200 SBT = PAT(I)  
210 SCT = SBT + PST(I)  
220 NPD = NPD + 1  
230 BTPU = BTPU + PST(I)  
240 I = I + 1  
250 GOTO 320  
260 IDT = NPAT(J)  
270 SBT = NPAT(J)  
280 SCT = SBT + NPST(J)  
290 NNPd = NNPd + 1  
300 BTNPU = BTNPU + NPST(J)  
310 J = J + 1  
320 IF SET >= T THEN 550  
330 IF PAT(I) > SET THEN 400  
340 SBT = SCT  
350 SCT = SBT + PST(I)  
360 NPD = NPD + 1  
370 BTPU = BTPU + PST(I)  
380 I = I + 1  
390 GOTO 530  
400 IF NPAT(J) > SET THEN 450
```

```
410 SBT = SCT
420 SCT = SBT + NPST(J)
430 J = J + 1
440 GOTO 530
450 IF PAT(I) > NPAT(J) THEN 500
460 IDT = PAT(I) - SCT
470 SBT = PAT(I)
480 SCT = SBT
490 GOTO 530
500 IDT = NPAT(J) - SCT
510 SBT = NPAT(J)
520 SCT = SBT
530 GOTO 320
540 N = NPD + NNPD
550 MU1 = NPD/BTPC
560 MU2 = NNPD/BTNPC
570 LAM1 = NPA/T
580 LAM2 = NNPA/T
590 RHO1 = LAM1/MU1
600 RHO2 = LAM2/MU2
610 PRINT N, MU1, MU2, LAM1, LAM2, RHO1, RHO2
620 STOP
630 'processing part for c-server model'
640 INPUT "enter number of servers";C
650 I = 1: J = 1 : TIT = 0
660 FOR M = 1 TO C
670 NPD(M) = 0
680 NNPD(M) = 0
690 BTPU(M) = 0
700 BTNPU(M) = 0
710 NEXT M
720 FOR M = 1 TO C
730 IF PAT(I) > NPAT(J) THEN 810
740 IDT(M) = PAT(I)
750 SBT(M) = PAT(I)
760 SCT(M) = SBT(M) + PST(I)
770 NPD(M) = NPD(M) + 1
780 BTPU(M) = BTPU(M) + PST(I)
790 I = I + 1
800 GOTO 870
```

```
810 IDT(M) = NPAT(J)
820 SBT(M) = NPAT(J)
830 SCT(M) = SBT(M) + NPST(J)
840 NNPD(M) = NNPD(M) + 1
850 BTNPU(M) = BTNPU(M) + NPST(J)
860 J = J + 1
870 NEXT M
880 'largest set(m) ,m = 1 to c is obtained'
890 GOSUB 1340
900 IF SCT(L) >= T THEN 1180
910 'smallest sct(m) . m = 1 to c is obtained '
920 GOSUB 1420
930 IF PAT(I) > SCT(S) THEN 1000
940 SBT(N) = SCT(N)
950 SCT(N) = SBT(N) + PST(I)
960 NPD(N) = NPD(N) + 1
970 BTPU(N) = BTPU(N) + PST(I)
980 I = I + 1
990 GOTO 1170
1000 IF NPAT(J) > SCT(N) THEN 1070
1010 SBT(N) = SCT(N)
1020 SCT(N) = SBT(N) + NPST(J)
1030 NNPD(N) = NNPD(N) + 1
1040 BTNPU(N) = BTNPU(N) + NPST(J)
1050 J = J + 1
1060 GOTO 1170
1070 IF PAT(I) > NPAT(J) THEN 1130
1080 IDT(N) = PAT(I) - SCT(N)
1090 TIDT(N) = TIDT(N) + IDT(N)
1100 SBT(N) = PAT(I)
1110 SCT(N) = SBT(N)
1120 GOTO 1170
1130 IDT(N) = NPAT(J) - SCT(N)
1140 TIDT(N) = TIDT(N) + IDT(N)
1150 SBT(N) = NPAT(J)
1160 SCT(N) = SBT(N)
1170 GOTO 880
1180 NPD = 0 : BTPC = 0
1190 NNPD = 0 : BTNPC = 0
1200 FOR M = 1 TO C
```

```
1210 NPD = NPD + NPD(M)
1220 BTPU = BTPU + BTPU(M)
1230 NNPД = NNPД + NNPД(M)
1240 BTNPU = BTNPU + BTNPU(M)
1250 NEXT M
1260 MU1 = NPD/BTPU
1270 MU2 = NNPД/BTNPU
1280 LAM1 = NPA/T
1290 LAM2 = NNPА/T
1300 RHO1 = LAM1/MU1
1310 RHO2 = LAM2/MU2
1320 PRINT T ,MU1 , MU2 , LAM1 ,LAM2 ,RHO1 , RHO2
1330 STOP
1340 'l , yielding largest sct(m) , m = 1 to c is obtained'
1350 L = SCT(1)
1360 FOR R = 1 TO C-1
1370 IF SCT(R+1) > L THEN 1390
1380 GOTO 1400
1390 L = SCT(R+1)
1400 NEXT R
1410 RETURN
1420 's , yielding smallest sct(m) , m = 1 to c is obtained'
1430 S = SCT(1)
1440 FOR R = 1 TO C-1
1450 IF SCT(R+1) < S THEN 1470
1460 GOTO 1480
1470 S = SCT(R+1)
1480 NEXT R
1490 RETURN
1500 'arrival time modules'
1510 INPUT "enter priority arrival rate " ; LAM1
1520 GOSUB 1560
1530 INPUT "enter nonpriority arrival rate "; LAM2
1540 GOSUB 1710
1550 RETURN
1560 'arrival time module-1 '
1570 I = 1 : NPA = 0 : AT = 0
1580 OPEN "o" ,1 , "arri.dat"
1590 RANDOMIZE
1600 R = RND * 100
```

```
1610 X = -LOG(1-(R/100)) * (1/LAM1)
1620 AT = AT +X
1630 IF AT >= T THEN 1690
1640 PRINT "arr1.dat" ; AT
1650 WRITE #1 , AT
1660 I = I + 1
1670 NPA = NPA + 1
1680 GOTO 1600
1690 CLOSE 1
1700 RETURN
1710 'arrival module -2'
1720 J = 1 : NNPA = 0 : AT = 0
1730 OPEN "o" , 2 , "arr2.dat"
1740 RANDOMIZE
1750 R = RND * 100
1760 Y = -LOG(1-(R/100)) * (1/LAM2)
1770 AT = AT + Y
1780 PRINT "arr2.dat" ; AT
1790 IF AT >= T THEN 1840
1800 WRITE #2 , AT
1810 J = J + 1
1820 NNPA = NNPA + 1
1830 GOTO 1750
1840 CLOSE 2
1850 RETURN
1860 'service time calculation modules'
1870 INPUT "enter priority service rate " ; MU1
1880 GOSUB 1920
1890 INPUT "enter nonpriority service rate " ; MU2
1900 GOSUB 2040
1910 RETURN
1920 'service time module -1'
1930 I = 1
1940 OPEN "o" ,3 , "ser1.dat"
1950 RANDOMIZE
1960 R = RND * 100
1970 X = -LOG(1-(R/100)) * (1/MU1)
1980 PRINT "ser1.dat" ; X
1990 WRITE #3, X
2000 I = I + 1
```

```
2010 IF I <= NPA THEN 1960
2020 CLOSE 3
2030 RETURN
2040 'service time module - 2 '
2050 J = 1
2060 OPEN "o" , 4 , "ser2.dat"
2070 RANDOMIZE
2080 R = RND *100
2090 Y = -LOG(1-(R/100)) * (1/MU2)
2100 WRITE #4 , Y
2110 PRINT "ser2.dat"; Y
2120 J = J + 1
2130 IF J <= NNPA THEN 2080
2140 CLOSE 4
2150 RETURN
2160 OPEN "i" , 1 , "arr1.dat"
2170 OPEN "i" , 2 , "arr2.dat"
2180 I = 1 : J = 1
2190 IF EOF(1) THEN 2260
2200 INPUT #1 , A
2210 PRINT "From arr1.dat" ; A
2220 PAT(I) = A
2230 I = I + 1
2240 IF EOF(1) THEN 2260
2250 GOTO 2190
2260 CLOSE 1
2270 IF EOF(2) THEN .2330
2280 INPUT #2 , B
2290 PRINT "From arr2.dat" ; B
2300 NPAT(I) = B
2310 J = J + 1
2320 GOTO 2270
2330 CLOSE 2
2340 RETURN
2350 OPEN "i" , 3 , "ser1.dat"
2360 I = 1 : J = 1
2370 INPUT #3 , C
2380 PRINT "From ser1.dat" ; C
2390 PST(I) = C
2400 I = I + 1
```

```
2400 I = I + 1
2410 IF EOF(3) THEN 2430
2420 GOTO 2370
2430 CLOSE 3
2440 OPEN "i" , 4 , "ser2.dat"
2450 IF EOF(4) THEN 2510
2460 INPUT #4 , D
2470 PRINT "From ser2.dat
2480 NFST(J) = D
2490 J = J + 1
2500 GOTO 2450
2510 CLOSE 4
2520 RETURN
```



```

10 INPUT "enter cs , c1 , c2 values " ; CS , C1 , C2
20 INPUT "ENTER LOWER LIMIT FOR B1 " ; B11
30 INPUT "ENTER UPPER LIMIT FOR B1" ; B12
40 INPUT "ENTER LOWER LIMIT FOR B2" ; B21
50 INPUT "ENTER LOWER LIMIT FOR B2" ; B22
60 INPUT "ENTER RANGE FOR B1 " ;R1
70 INPUT "ENTER RANGE FOR B2 " ;R2
80 FOR B1 = B11 TO B12 STEP R1
90 FOR B2 = B21 TO B22 STEP R2
100 PRINT "present set of b1 , b2 values" ; B1 , B2
110 A = -4 *B2*C2
120 B = 12*B1*B2 * (2*B2+C2)
130 C = 12*B1*B2* (4*B1*B2* + 2*B1*C2 - 3*B2 )
140 D = 12*B1^2*B2 * (4*B1*B2 + 2*B1*C2 - 24*B2)
150 E = -648*B1^2 * B2^2
160 PRINT "guessing preferable initial value x0 "
170 PRINT
180 INPUT "ENTER LOWER LIMIT FOR X0" ; D1
190 INPUT "ENTER UPPER LIMIT FOR X0" ;D2
200 INPUT "ENTER RANGE FOR X0" ; R3
210 FOR Y = D1 TO D2 STEP R3
220 FX0 = A*X0^4 + B*X0^3 + C*X0^2 + D*X + E
230 PRINT "value of x0 = " ; Y ; "value of f(x0) = " ; FX0
240 PRINT
250 INPUT "enter 1 to continue , 2 to stop searching " ; RESPONSE
260 IF RESPONSE = 1 THEN 270 ELSE 280
270 NEXT Y
280 INPUT "enter initial guess x0 " ; X0
290 X1 = X0
300 X0 = X1
310 F1 = A*X0^4 + B*X0^3 + C*X0^2 + D*X + E
320 F2 = 4*A*X0^3 + 3*B*X^2 + 2*C*X + D
330 X1 = X0 - ( F1 / F2 )
340 PRINT "x0 = " ; X0 ; "x1 = " ; X1
350 PRINT
360 INPUT "enter 1 to continue , 2 to stop iteration" ; RESPONSE2
370 IF RESPONSE2 = 1 THEN 300
380 PRINT "b1 = " ; B1 ; "b2 = " ; B2 ; "x1 = ";X1
390 PRINT

```

```
400 NEXT B2
410 NEXT B1
420 PRINT
430 PRINT "processing is over "
440 END
```