

プログラムのページ

担当和田英一

6804. Pattern Following Program

相馬 嵩 (理化学研究所情報科学研究室)

1. 目的 2次元格子上の黒白の pattern を follow して, X-Y plotter に表示するための, unit step vector の列を発生する.

2. 方法 2次元の pattern を端から走査して, 黒点に達すると, それに接続している黒点を follow してゆく. 分岐点があると, その点を覚えていて, 一方の枝を follow する. そして端点まできたら, 先程の分岐点まで戻って, 他の枝を follow する. この操作は recursive におこなう. follow した点は順に消してゆく. 接続しているすべての枝を follow し終ると, またはじめての点の次から走査をおこなう. 上の操作を, すべての点を走査し終るまでくり返す.

第1図に例を示す. 線に幅があって, 黒点が, と

りあっている場合も, 分岐点の特別な場合とみなされる.

3. プログラム (FORTRAN II¹⁾)

MAIN: test 用

RDPTRN: pattern をカードから読み込む.

FOLLOW: following をおこなう.

PLOT I: 整数座標の PLOT-routine

PE: picture element の黒白をしらべる.

P START

PLOT³⁾ } plotter control routine

P END

注) RDPTRN, PE では bitpattern の pack unpack のために assembler language²⁾ が使われている.

参考文献

- 1) OKITAC-5090 H FORTRAN-H(2) システム解説書 (41.7.16)
- 2) OKITAC-5090 H MASH-III (40.8.10)
- 3) Bulletin No. 170-E/Nov. 1966 Calcomp Inc.



第1図

```

C ***** PATTERN FOLLOWING MAIN *****
C DIMENSION IA(5000)
C COMMON ML,NL,IA,NW
C REWIND 2
C ***** READ PATTERN FROM CARD *****
C CALL RDPTRN
C ***** INITIALIZE PLOTTER CONTROL ROUTINE *****
C CALL PSTART(7HPATTERN)
C ***** SCAN POINTS *****
C DO 2 M=1,ML
C DO 2 N=1,NL
C ***** TEST BLACK OR WHITE *****
C IBW=-1
C CALL PE(M,N,IBW)
C IF(IBW) 2,2,1
C ***** FOLLOW BLACK POINT *****
C 1 CALL FOLLOW(M,N-1,M,N)
C 2 CONTINUE
C ***** FINALIZE PLOTTER CONTROL ROUTINE *****
C CALL PEND
C CALL EXIT
C END
C
C ***** SUBROUTINE FOLLOW *****
C SUBROUTINE FOLLOW(IM1,IN1,IM2,IN2)
C DIMENSION NPTM1(7),NPTN1(7)
C ***** INITIALIZE STACK POINTER *****
C IPOINT=1
C ***** SET PARAMETERS *****
C M1=IM1
C N1=IN1
C M2=IM2
C N2=IN2
C ***** MOVE PEN TO BRANCH POINT AND ERASE *****
C 1 IPEN=3
C 2 CALL PLOTI(M2,N2,IPEN)
C CALL PE(M2,N2,0)
C MD=M2-M1
C ND=N2-N1
C IF(MD*ND) 4,3,4
C 3 IF(XABSF(MD)+XABSF(ND)) 5,26,5
C 4 ISW=1
C GO TO 6
C 5 ISW=2
C 6 NNEXT=0
C ***** SELECT NEAREST NEIGHBOR IN PROBABLE
C ORDER *****
C DO 24 I=1,7
C GO TO ( 7,15),ISW
C 7 GO TO ( 8, 9,10,11,12,13,14),I

```

```

      8 MD1=MD
      ND1=ND
      GO TO 22
      9 MD1=MD
      ND1=0
      GO TO 22
     10 MD1=0
      ND1=ND
      GO TO 22
     11 MD1=-MD
      ND1=ND
      GO TO 22
     12 MD1=MD
      ND1=-ND
      GO TO 22
     13 MD1=-MD
      ND1=0
      GO TO 22
     14 MD1=0
      ND1=-ND
      GO TO 22
     15 GO TO ( 8,16,17,18,19,20,21),I
     16 MD1=MD*XABSF(MD)+(1-XABSF(MD))
      ND1=ND*XABSF(ND)+(1-XABSF(ND))
      GO TO 22
     17 MD1=MD*XABSF(MD)-(1-XABSF(MD))
      ND1=ND*XABSF(ND)-(1-XABSF(ND))
      GO TO 22
     18 MD1=(1-XABSF(MD))
      ND1=(1-XABSF(ND))
      GO TO 22
     19 MD1=-(1-XABSF(MD))
      ND1=-(1-XABSF(ND))
      GO TO 22
     20 MD1=-MD*XABSF(MD)+(1-XABSF(MD))
      ND1=-ND*XABSF(ND)+(1-XABSF(ND))
      GO TO 22
     21 MD1=MD*XABSF(MD)-(1-XABSF(MD))
      ND1=-ND*XABSF(ND)-(1-XABSF(ND))
      GO TO 22
      C ***** SET COORDINATE OF NEXT POINT *****
      NPTM=M2+MD1
      NPTN=M2+ND1
      C ***** TEST BLACK OR WHITE *****
      IBW=-1
      CALL PE(NPTM,NPTN,IBW)
      IF(IBW) 24,24,23
      C ***** SAVE IF BLACK AND COUNT *****
      23 NNEXT=NNEXT+1
      NPTM(NNEXT)=NPTM
      NPTN(NNEXT)=NPTN
      C ***** SET MOST PROBABLE ALLOWABLE NEXT POINT *****
      24 CONTINUE
      ***** TEST NUMBER OF BLACK POINT *****
      IF(NNEXT-1) 25,28,29
      C ***** TEST IF STACK IS EMPTY *****
      25 IF(IPOINT-1) 26,26,27
      26 RETURN
      C ***** PUSH UP STACK AND SET PARAMETERS FOR
      C ***** STARTING POINT *****
      27 IPOINT=IPOINT-1
      BACK SPACE 2
      READ TAPE 2, M1,N1,M2,N2
      BACK SPACE 2
      GO TO 1
      C ***** BLACK POINT ONLY ONE *****
      28 M1=M2
      N1=N2
      M2=NPTM(1)
      N2=NPTN(1)
      IPEN=2
      GO TO 2
      C ***** BLACK POINT MORE THAN ONE *****
      29 NNEXT=0
      NNEXTA=0
      C ***** TEST AND COUNT ALLOWABLE BLACK POINT **
      DO 32 I=1,NNEXT
      ICM=NPTM(I)+(NPTM(I)-M2)
      ICN=NPTN(I)+(NPTN(I)-N2)
      IBW=-1
      CALL PE(ICM,ICM,IBW)
      IF(IBW) 32,32,30
      30 NNEXTA=NNEXTA+1
      IF(NEXTA) 32,31,32
      31 NEXTA=I
      32 CONTINUE
      IF(NNEXTA-1) 33,34,34
      C ***** SET MOST PROBABLE NEXT POINT IF NO
      C ***** ALLOWABLE POINT EXISTS *****
      IWM=NPTM(1)
      IWN=NPTN(1)
      GO TO 35
      C ***** SET MOST PROBABLE ALLOWABLE NEXT POINT
      34 IWM=NPTM(NNEXTA)
      IWN=NPTN(NNEXTA)
      C ***** PUSH DOWN COORDINATES OF BRANCH POINT *
      35 WRITE TAPE 2,M1,N1,M2,N2
      IPOINT=IPOINT+1
      C ***** SET PARAMETERS FOR ONE BRANCH *****
      M1=M2
      N1=N2

```

```

M2=IWM
N2=IWN
***** MOVE PEN *****
IPEN=2
GO TO 2
END

C ***** SUBROUTINE PLOTI *****
C SUBROUTINE PLOTI(I,J,IPEN)
C X=FLOAT(I)*0.02
C Y=FLOAT(J)*0.02
C CALL PLOT(X,Y,IPEN)
C RETURN
C END

C ***** SUBROUTINE PE *****
C SUBROUTINE PE(M,N,IBW)
C DIMENSION IA(5000)
C COMMON ML,NL,IA,NW
C IF(N) 3,3,1
C 1 IF(N) 3,3,2
C 2 MN=(M-1)*NW+(N-1)/42
C ISH=42-XMODF(N-1,42)
C ***** UNPACK PICTURE ELEMENT *****

C M L,XR1/MN
C L,AR1/IA,XR1
C SRL,2,AR1/(ISH)
C L,AR3/(IBW)
C JZ,AR3/SET0
C JP,AR3/SET1
C ***** SET OUTPUT VARIABLE 1 OR 0 *****
C LLI,AR1/
C SLL,2,AR1/1
C T,AR1/(IBW)
C J/RET
C ***** SET WHITE *****
C SET0 RB,AR2/1,42
C ***** SET BLACK *****
C SET1 SB,AR2/1,42
C SLL,2,AR1/(ISH)
C T,AR1/IA,XR1
C RET NE/
C M 3 RETURN
C END

C ***** SUBROUTINE ROPTRN *****
C SUBROUTINE ROPTRN

```

```

DIMENSION IDATA(70),IA(5000)
COMMON ML,NL,IA,NW
***** READ PARAMETER CARD *****
READ INPUT TAPE 1,1, ML,NL
1 FORMAT(2I4)
NW=(NL-1)/42+1
I=0
***** READ PATTERN FROM CARD AND PACK THEM
***** INTO WORDS *****
DO 5 L=1,ML
NL1=NL
6 READ INPUT TAPE 1,2, IDATA
2 FORMAT(70I1)
DO 3 J=1,5
I=I+1
IA(I)=0
DO 5 K=1,14
JK=(J-1)*14+K
L,XR1/I
L,XR2/JK
L,AR1/IA-1,XR1
SLL,1,AR1/3
ASP,AR1/IDATA-1,XR2
T,AR1/IA-1,XR1
NL1=NL1-3
IF(NL1) 4,4,3
3 CONTINUE
GO TO 6
4 IS=(14-K)*3
L,XR1/I
L,AR1/IA-1,XR1
SLL,1,AR1/(IS)
T,AR1/IA-1,XR1
5 CONTINUE
RETURN
END

```