ALGORITHM 278
GRAPH PLOTTER [J6]
P. LLOYD (Recd. 4 June 1965)
Queen Mary College, London, England

```
procedure graphplotter (N, x, y, m, n, xerror, yerror, g, L, S, EM,
  C0, C1, C2, C3, C4, label);
  value N, m, n, xerror, yerror, g, L, S;
  array x, y;
  integer N, g, m, n, L, S;
  real xerror, yerror;
  string EM, C0, C1, C2, C3, C4;
  label label;
comment This procedure is intended to be used to give an ap-
  proximate graphical display of a multivalued function, y[i, j] of
  x[i], on a line printer. Output channel N is selected for all out-
  put from graphplotter. The display is confined to points for which
  1 ≤ i ≤ m and 1 ≤ j ≤ n where 2 ≤ n ≤ 4. If n = 1, then y is
  considered to be a one-dimensional array y[i] and the display is
  again given for 1 ≤ i ≤ m. The format of the print out is ar-
  ranged so that a margin of g spaces separates the display from
  the left-hand side of the page. L and S denote the number of
  lines down the page and the number of spaces across the page
  which the display will occupy. The graph is plotted so that lines
  1 and L correspond to the minimum and maximum values of x,
  and the spaces 1 and S correspond to the minimum and maxi-
  mum values of y, that is, y is plotted across the page and x down
  the page. After the graph has been plotted, the ranges of x and
  y for which the display is given are printed out in the order as
  above, separated from the display by a blank line. The strings
  EM ··· C4 must be such that they occupy only one character
  position when printed out. The characters of C1 C2 C3 C4 repre-
  sent y[i,1] y[i,2] y[i,3] y[i,4]. EM is the character printed out
  round the perimeter of the display. C0 is printed at empty
  positions. At coincident points the order of precedence of the
  characters is C1 C2 C3 C4 EM C0. For the special case n=1 the
  character C1 represents y[i]. Control is passed from the pro-
  cedure to the point labeled label if the interval between the
  maximum value and minimum values of x[i] is less than xerror,
  or if the range of y is less than yerror. If the values of x[i] occur
  at equal intervals, choosing L=m will make one line equivalent
  to one interval of x;
begin
  real p, q, xmax, xmin, ymax, ymin;
  integer i, j;
  integer array plot[1:L,1:S];
  xmax := xmin := x[1];
  for i := 2 step 1 until m do
  begin
    if x[i] > xmax then xmax := x[i];
    if x[i] < xmin then xmin := x[i]
  end of hunt for maximum and minimum values of x;
  if n=1 then go to N1A;
  ymax := ymin := y[1,1];
  for i := 1 step 1 until m do
    for j := 1 step 1 until n do
    begin
      if y[i,j] > ymax then ymax := y[i,j];
      if y[i,j] < ymin then ymin := y[i,j]
    end of hunt for maximum and minimum values of y;
escape: if abs(xmax−xmin) < xerror ∨ abs(ymax−ymin) <
  yerror then go to label;
  p := (L−1)/(xmax−xmin); q := (S−1)/(ymax−ymin);
  for i := 1 step 1 until L do
    for j := 1 step 1 until S do plot[i,j] := 2;
  for i := 1, L do
    for j := 1 step 1 until S do plot[i,j] := 1;
  for i := 2 step 1 until L−1 do
    for j := 1, S do plot[i,j] := 1;
  if n = 1 then go to N1B;
  for i := 1 step 1 until m do
    for j := n step −1 until 1 do
      plot[1+entier(0.5+p×(x[i]−xmin)),
        1+entier(0.5+q×(y[i,j]−ymin))] := j+2;
plotter:
  for i := 1 step 1 until L do
  begin
    NEWLINE(N,1); SPACE(N,g);
    comment NEWLINE and SPACE must be declared
      globally to graphplotter, NEWLINE(N,p) outputs p car-
      riage returns and p line feeds on channel N, SPACE(N,p)
      outputs p blank character positions on channel N;
    for j := 1 step 1 until S do
    begin
      switch SW := SW1, SW2, SW3, SW4, SW5, SW6;
      go to SW[plot[i,j]];
SW1: outstring(N,EM); go to fin;
SW2: outstring(N,C0); go to fin;
SW3: outstring(N,C1); go to fin;
SW4: outstring(N,C2); go to fin;
SW5: outstring(N,C3); go to fin;
SW6: outstring(N,C4);
fin:
    end
  end of display output;
  NEWLINE(N,2); SPACE(N,g); outreal(N,xmin);
    outreal(N,xmax);
  outreal(N,ymin); outreal(N,ymax);
  go to end;
N1A:
  ymax := ymin := y[1];
  for i := 2 step 1 until m do
  begin
    if y[i] > ymax then ymax := y[i];
    if y[i] < ymin then ymin := y[i]
  end of hunt for maximum and minimum values of y when
    n = 1;
  go to escape;
N1B:
  for i := 1 step 1 until m do
    plot[1+entier(0.5+p×(x[i]−xmin)),
      1+entier(0.5+q×(y[i]−ymin))] := 3;
  go to plotter;
end:
end of graphplotter
```

1966 CONFERENCE DATES

ACM SYMSAM	March 29–31	WASHINGTON
SPRING JCC	April 26–28	BOSTON
ACM 66	August 30–Sept. 1	LOS ANGELES
FALL JCC	November 8–10	SAN FRANCISCO