```
p,algol,1<
PROGRAM TIL INVERSION AF LAPLACE TRANSFORM 120365 edda sveinsdottir
dog først
Procedurer til styring af CALCOMP-plotter.
GA4 version: Mogens Kjær
begin
   integer xxx,yyy; real deltax,deltay;
   boolean penstate;

   procedure plotter(xx,yy,pen);
   value xx,yy,pen;
   integer xx,yy;
   boolean pen;
   begin
      integer s;
      s:=select(64);
      if -,(pen = penstate) then
      begin
         penstate:=pen;
         writechar(if pen then 32 else 16)
      end change pen state;
      code xx, yy;
      3, 44;
      3, 44;
   xrn      0 , ann   pa1  ;  M := 0, R := abs(xx)
   sn     pa2 , cl      1  ;  - abs(yy)
   arn    pa1 , cl      1  ;  xx
   arn    pa2 , cl      1  ;  yy
   cln     -8 , ga    re6  ;  store bit×4
   arn    re7 , pm    re8  ;  fetch parameters
e6:cl       0 , mb    re1  ;  move bits
   gaX    re3              ;  store straight step
   mb     re1 , ga    re4  ;  store diagonally step
   annX   pa1              ;  M := abs(xx)
   ann    pa2 , tl      1  ;  R := abs(yy), M := M×2, R := R×2
   bs   (re6) t +12        ;  shift M and R
   xr       0              ;  if abs(yy) > abs(xx)
   gr     pa2 , gm    pa1  ;  store yy and xx
   srn    pa1 , tl     -1  ;  test := - xx:2
e2:hv     re1 , ar    pa2  ;  test := test + yy
e3:syV      0          LT  ;  straight step
e4:sy       0 , sr    pa1  ;  diagonally step
e1:xr      15 , sr    c42  ;  xx := xx - 1
   hhX    re2          NT  ;  xx ≥ 0
   hv     re5
e7:qq  4.9+4.13+ 2.17+2.21+ 8.25+1.29+ 8.33+1.37  ;  parameter 1
e8:qq 12.9+5.13+10.17+3.21+12.25+5.29+10.33+3.37  ;  parameter 2
e5:   e;
      select(s)
   end plotter;

   procedure plotgraph(x0,x1,x,y,dx);
   value x0,x1; real x0,x1,x,y,dx;
   begin
      integer xx,yy,s;
      x:=x0; xx:=x0/deltax-xxx; yy:=y/deltay-yyy;
      plotter(xx,yy,false);
      xxx:=xxx+xx; yyy:=yyy+yy;
      for x:=x0 step dx until x1 do
      begin
         xx:=x/deltax-xxx; yy:=y/deltay-yyy;
         plotter(xx,yy,true);
         xxx:=xxx+xx; yyy:=yyy+yy
      end
```

```
end plotgraph;

procedure plotline(x0,y0,x1,y1);
value x0,y0,x1,y1;
real x0,y0,x1,y1;
begin
   integer xx,yy;
   xx:=x0/deltax-xxx; yy:=y0/deltay-yyy;
   plotter(xx,yy,false);
   xxx:=xxx+xx; yyy:=yyy+yy;
   xx:=x1/deltax-xxx; yy:=y1/deltay-yyy;
   plotter(xx,yy,true);
   xxx:=xxx+xx; yyy:=yyy+yy
end plotline;

real procedure plotchar(t,x,y,h);
value t,x,y,h;
integer t;
real x,y,h;
begin
   integer xx,yy,n1,n2,hh,i,x1,y1;
   boolean hop,AA,flyt,A0,A1;
   if t≥58 then goto exit; hh:=h×20;
   if t<29 then
   begin
      case (t+1) of
      begin
         begin comment t=0;
            A0 := 10 0 10 0 10 0 10 0;
            A1 := 10 0 10 0 10 0 10 0;
         end;
         begin comment t=1;
            A0 := 10 106 10 556 10 0 10 0;
            A1 := 10 0 10 0 10 0 10 0;
         end;
         begin comment t=2;
            A0 := 10 228 10 369 10 704 10 595;
            A1 := 10 0 10 0 10 0 10 0;
         end;
         begin comment t=3;
            A0 := 10 261 10 751 10 659 10 390;
            A1 := 10 32 10 0 10 0 10 0;
         end;
         begin comment t=4;
            A0 := 10 209 10 84 10 463 10 384;
            A1 := 10 0 10 0 10 0 10 0;
         end;
         begin comment t=5;
            A0 := 10 311 10 163 10 500 10 620;
            A1 := 10 193 10 0 10 0 10 0;
         end;
         begin comment t=6;
            A0 := 10 369 10 356 10 38 10 403;
            A1 := 10 655 10 290 10 0 10 0;
         end;
         begin comment t=7;
            A0 := 10 101 10 742 10 0 10 0;
            A1 := 10 0 10 0 10 0 10 0;
         end;
         begin comment t=8;
            A0 := 10 457 10 139 10 566 10 489;
            A1 := 10 65 10 204 10 628 10 480;
         end;
         begin comment t=9;
```

```
   A0 := 10 405 10 456 10 100 10 369;
   A1 := 10 723 10 390 10 32 10 0;
end;
begin comment t=10;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=11;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=12;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=13;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=14;
   A0 := 10 88 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=15;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 1001 10 336 10 489 10 297;
end;
begin comment t=16;
   A0 := 10 289 10 139 10 566 10 620;
   A1 := 10 193 10 0 10 0 10 0;
end;
begin comment t=17;
   A0 := 10 115 10 277 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=18;
   A0 := 10 321 10 204 10 628 10 100;
   A1 := 10 369 10 704 10 0 10 0;
end;
begin comment t=19;
   A0 := 10 228 10 189 10 927 10 556;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=20;
   A0 := 10 197 10 38 10 403 10 736;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=21;
   A0 := 10 101 10 413 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=22;
   A0 := 10 165 10 207 10 605 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=23;
   A0 := 10 178 10 191 10 736 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=24;
   A0 := 10 165 10 477 10 460 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=25;
```

```
         A0 := 10 196 10 183 10 18 10 608;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=26;
         A0 := 10 0 10 0 10 0 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=27;
         A0 := 10 262 10 287 10 398 10 658;
         A1 := 10 384 10 0 10 0 10 0;
      end;
      begin comment t=28;
         A0 := 10 0 10 0 10 0 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
   end
end
else
begin
   case (t-28) of
   begin
      begin comment t=29;
         A0 := 10 0 10 0 10 0 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=30;
         A0 := 10 0 10 0 10 0 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=31;
         A0 := 10 0 10 0 10 0 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=32;
         A0 := 10 168 10 671 10 431 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=33;
         A0 := 10 193 10 204 10 631 10 544;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=34;
         A0 := 10 192 10 162 10 745 10 576;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=35;
         A0 := 10 133 10 18 10 608 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=36;
         A0 := 10 160 10 174 10 952 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=37;
         A0 := 10 128 10 178 10 736 10 0;
         A1 := 10 0 10 0 10 0 10 0;
      end;
      begin comment t=38;
         A0 := 10 289 10 139 10 566 10 620;
         A1 := 10 193 10 0 10 0 10 0;
      end;
      begin comment t=39;
         A0 := 10 224 10 177 10 725 10 450;
         A1 := 10 0 10 0 10 0 10 0;
```

```
end;
begin comment t=40;
   A0 := 10 449 10 139 10 566 10 620;
   A1 := 10 193 10 1005 10 594 10 576;
end;
begin comment t=41;
   A0 := 10 288 10 177 10 725 10 450;
   A1 := 10 466 10 0 10 0 10 0;
end;
begin comment t=42;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=43;
   A0 := 10 385 10 139 10 566 10 620;
   A1 := 10 193 10 992 10 736 10 0;
end;
begin comment t=44;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=45;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=46;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=47;
   A0 := 10 0 10 0 10 0 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=48;
   A0 := 10 352 10 139 10 945 10 450;
   A1 := 10 846 10 408 10 0 10 0;
end;
begin comment t=49;
   A0 := 10 192 10 568 10 1000 10 640;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=50;
   A0 := 10 352 10 177 10 719 10 111;
   A1 := 10 659 10 384 10 0 10 0;
end;
begin comment t=51;
   A0 := 10 278 10 555 10 129 10 204;
   A1 := 10 608 10 0 10 0 10 0;
end;
begin comment t=52;
   A0 := 10 224 10 177 10 723 10 384;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=53;
   A0 := 10 247 10 163 10 483 10 18;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=54;
   A0 := 10 215 10 163 10 483 10 0;
   A1 := 10 0 10 0 10 0 10 0;
end;
begin comment t=55;
   A0 := 10 342 10 555 10 129 10 204;
   A1 := 10 628 10 256 10 0 10 0;
```

```
            end;
            begin comment t=56;
               A0 := 10 192 10 163 10 695 10 576;
               A1 := 10 0 10 0 10 0 10 0;
            end;
            begin comment t=57;
               A0 := 10 203 10 753 10 390 10 576;
               A1 := 10 0 10 0 10 0 10 0;
            end
         end
      end;
      AA:=A0;
      n1:=integer ((AA shift -35)∧35 0 5 m);
      y1:=integer ((AA shift -30)∧35 0 5 m);
      x1:=y1:6;
      y1:=(y1-x1×6)×hh; x1:=x1×hh;
      xx:=x1+x/deltax-xxx; yy:=y1+y/deltay-yyy;
      xxx:=x/deltax; yyy:=y/deltay;
      plotter(xx,yy,false);
      n2:=(n1-8)×5; n1:=if n1≤7 then n1×5 else 35;
      flyt:=false;
      for i:=10 step 5 until n1,0 step 5 until n2 do
      begin
         if i=0 then AA:=A1;
         yy:=integer ((AA shift (i+4-39))∧35 0 5 m);
         xx:=yy:6;
         if yy=31 then
         begin
            flyt:=true;
         end
         else
         begin
            yy:=(yy-xx×6)×hh-y1;
            xx:=xx×hh-x1;
            plotter(xx,yy,-,flyt);
            flyt:=false;
            x1:=x1+xx; y1:=y1+yy
         end
      end i;
      xxx:=xxx+x1; yyy:=yyy+y1;
exit: plotchar:=if t<58 then x+100×h×deltax else x
   end plotchar;
   xxx:=yyy:=0; deltax:=deltay:=0.01;

   comment program af 120365 til inversion af Laplace transform = f(t)
   arbejder med 3 slags data:
      1)  syntetiske data af form f(t)=Sum[N[i]×exp(-lambda[i]×t)]
   som det selv genererer ved hjælp af real procedure f(x). I dette
   tilfælde skrives o på skrivemaskinen inden kompileringen, for at
   inkludere procedurekrop svarende til denne type data. pause-
   signalet ignoreres.
         2)  syntetiske data, som ikke er en sum af eksponential
   funktioner, men hvor f(t)=f(exp(x)) kan udtrykkes på explicit form.
   I dette tilfælde skrives der ikke o på skrivemaskinen, men når
   pause-signalet kommer, tastes procedurekroppen til real procedure
   f(x) ind fra skrivemaskinen. Eks:
             f:= exp(-0.001×exp(x))/(exp(x)+0.009)↑2.
         3)  experimentelle data i form af et array af funktionsværdier
   aflæst for ækvidistante værdier af lnt=x. I dette tilfælde tastes o
   før compileringen. Ved at gøre n negativ (se nedenfor) vil maskinen
   være parat til at indlæse en strimmel indeholdende data-array F[n,-n].
   pause-signalet ignoreres.
   DATASTRIMMEL I: maskinprocedure plot 111, 112, 113.
   DATASTRIMMEL II:bestaar af tabel over den komplekse funktion,
```

```
   Gamma(1+iu) hvor (u=0(0.1)10).
   INPUT FRA SKRIVEMASKINE:
   dx     skridtlængde ved x-integration  (f.eks. 0.25)
   du     skridtlængde ved u-integration, skal være mult af 0.1
   xmax   øvre grænse ved x-integrationen, tastes evt først ind , efter
          at plotteren har tegnet grafen af integranden ved x-integration
          skal være lige mult af dx
   umax   øvre grænse ved u-integration, skal være lige mult af du
   lamax  hhv øvre og nedre grænse for det interval for lamda, hvori
   lamin  man ønsker at undersøge g(lamda)/lamda. lamax og lamin skal
          hele mult af 10-i
   noofy  antal plottepunkter i det angivne lamda-interval
   n, n>0 antal exponentialled i den syntetiske kurve f(t) af typen 1)
          med parametrene
   N[i], lambda[i]   (i=1,...,n)
   n, n<0 i dette tilfælde indlæses strimmel med rigtige data
   F[n,-n] som er den experimentelle kurve aflæst med ækvidistancen dx
          i x-intervallet [n×dx,-n×dx];

   begin
      integer n,m,noofy,Kplads;
      boolean nykurve;
      real dx,xmax,du,umax,lamax,lamin;
      array Kc,Ks[0:100];

      procedure plotcond(x,y,t,e1,e2,b1,b2);
      real x,y,t,e1,e2;
      boolean b1,b2;
      begin
         integer tr,xx,yy;
         boolean hop,s;
         s:= true;
         for t:= e1,e2 while b1 do
         if -, b2 then s:= true else
         begin
            xx:= x/deltax - xxx; yy:= y/deltay - yyy;
            if s then
            begin
               plotter(xx,yy,false);
               xxx:= xxx+xx; yyy:= yyy+yy;
               s:= false
            end
            else
            begin
               plotter(xx,yy,true);
               xxx:= xxx+xx; yyy:= yyy+yy
            end
         end
      end plotcond;

      select(64);
      writechar(16);
      penstate:=false;
      select(16);
      for n:=0 step 1 until 100 do
      begin
         Kc[n] := read real;
         Ks[n] := read real
      end n;
      n:= 0;
DATA:
      select(17);
      writecr;  writetext(ǂ<dx,xmax,du,umax,lamax,lamin,noofy:
ǂ);
```

```
      select(33);
      writecr;
      writetext(§<program 120365 - inversion af Laplace transform§);
      dx:= read real;
      xmax:= read real;
      du:= read real;
      umax:= read real;
      lamax:= read real;
      lamin:= read real;
      noofy:= read integer;
      m:= 1;
      select(17);
      writecr;
      writetext(§<m:§);
      m:= read integer;
      if m=0 then goto NÆSTYDERSTE BLOK;
      if m<0 then
      begin
         n:= -m;
         nykurve:= true;
         goto NÆSTYDERSTE BLOK
      end;
      n:= m;
      writecr;
      writetext(§<N(i), lambda(i): §);
      nykurve:= true;

NÆSTYDERSTE BLOK:
      begin
         integer i,j,k,uo,us,K,S,grænse;
         real x,u,y,ymax,ymin,dy,gy,pi,Fcu,Fsu,Kcu,Ksu,nævner,lamda,lstep,gmax;
         array N,lambda[1:if n=0 then 1 else n],g[0:noofy],F[-n:if n=0 then 1 else n]
         real procedure f(x);
         value x;
         real x;
copy t<
         if m<0 then
         begin
            select(32);
            for i:=-n step 1 until n do F[i]:=read real;
            select(33);
            writecr;
            writetext(§<real data§);
            goto XINT
         end;
         select(33);
         if m>0 then
         for i:=1 step 1 until n do
        begin
           N[i]:= read real;
           lambda[i]:= read real
        end;
         select(33);
         writecr;
         writecr;
         writetext(§<N(i), lambda(i): §);
         for i:=1 step 1 until n do
         begin
            writecr;
            write(§-ddddd.ddd§,N[i],lambda[i])
         end i;
         if m=0 then goto XINT;

         begin
```

```
            real DX;
            comment i denne blok bliver log y, hvor y=exp(x)×f(x), plottet mod x
(x=-10,-9,...,10) i skridt af o.25. x-enheden=0.25 tomme. y-dekaden
=1.5 tomme. Der er 5 dekader;
            deltax:= 0.04;  deltay:= 0.02/3×2.302585;
            select(17);
            comment faktoren 2.302585 i deltay er lig ln 10. Dvs kan ved kald af
plotprocedurer benytte ln i stedet for log10;
            writetext(ǂ<
GØR CALCOMP KLAR t×f(t) plottes mod ln(t) - hvis dette uønsket tast 0: ǂ);
            j:= read integer;
            if j=0 then goto XINT;
            x:= -10;
            DX:= 0.4;
            xxx:= yyy:= 0;
YAXE:
            for j:=1,10,100,1000,10000 do
            for k:=2 step 1 until 10 do
            begin
               plotline(x,ln((k-1)×j),x,ln(k×j));
               plotline(x,ln(k×j),x+DX,ln(k×j));
            end k;
            if x=10 then goto XAXE;
            for k:=6 step -1 until 1 do
            begin
               x:= -10-k×0.7;
               j:= 1;
               for j:=j while x<-10.6 do
               begin
                  x:= plotchar(j,x,ln(10↑(k-1)),0.175);
                  j:= 16
               end;
            end k;
            x:= 10;
            DX:= -0.4;
            goto YAXE;
XAXE:
            deltay:= deltay/2.302585×1.5;
            comment y-enhed = 1 tomme;
            for j:=8 step -2 until -10 do
            begin
               plotline(j+2,0,j,0);
               plotline(j,0,j,-0.1);
               if j=0 then
               begin
                  plotchar(16,-0.24,-0.4,0.15);
                  x:= plotchar(23,-1.5,-0.9,0.15);
                  plotline(x+0.16,-0.8,x+0.44,-0.8);
                  plotline(x+0.16,-0.85,x+0.44,-0.85);
                  x:= x+0.6;
                  for i:=35,37,19 do x:= plotchar(i,x,-0.9,0.15);
               end
               else
               if j>0 then plotchar(j,j-0.24,-0.4,0.15)
               else
               if jǂ-10 then
               begin
                  plotline(j-0.8,-0.325,j-0.55,-0.325);
                  plotchar(-j,j-0.24,-0.4,0.15)
               end;
            end;
            x:= -8;
            x:= plotchar(53,x,7,0.15);
            x:= plotchar(23,x,7.1,0.1);
```

```
            for j:=54,53 do x:= plotchar(j,x,7,0.15);
            plotchar(23,x,7.1,0.1);
            deltay:= deltay×2.302585/1.5;
            comment y-dekade = 1.5 tomme;
            plotcond(x,x+ln(f(x)),x,-10.0,x+0.15,x≤10,x+ln(f(x))>ln(0.5));
            select(17);
            writecr;
            writetext(‡<xmax (lige mult af dx): ‡);
            xmax:= read real;
         end paa blok med plotning af exp(x)×f(x);

XINT:
         grænse:= entier(xmax/dx);
         j:= entier(umax×10);
         begin
            real k1,k2,k3,k4,k5;
            array Fc,Fs[0:j];
            integer u1,du1,umax1;
            procedure Kcs(t,i);
            value i;
            integer i;
            string t;
            begin
               integer s;
               if kbon then
               begin
                  s:=select(33);
                  writecr;
                  writetext(‡<Kcs: ‡);
                  writetext(t);
                  write(‡-dddddd‡,i);
                  select(s)
               end
            end;
            begin
               real plus,minus,Fcos,Fsin;
               array fc,fs[0:grænse];
               comment i denne blok foretages x-integrationen ved hjælp af
Simpson. Dvs Fc(u) og Fs(u), hhv den reelle og imaginære del
af F(u) beregnes;
               for k:=0 step 1 until grænse do
               begin
                  x:= k×dx;
                 if m<0 then
                 begin
                    plus:= exp(x)×F[k];
                    minus:= exp(-x)×F[-k]
                  end
                  else
                  begin
                     plus:= exp(x)×f(x);
                     minus:= exp(-x)×f(-x)
                  end;
                  fc[k]:= plus+minus;
                  fs[k]:= plus-minus;
               end k;
               du1:= du×10;
               umax1:= umax×10;
               for u1:=0 step du1 until umax1 do
               begin
                  u:= u1/10;
                  Fcos:= fc[0]+4×fc[1]×cos(u×dx)+fc[grænse]×cos(u×xmax);
                  Fsin:= 4×fs[1]×sin(u×dx)+fs[grænse]×sin(u×xmax);
                  for k:=2 step  2 until grænse-2 do
```

```
                    begin
                       Fcos:= Fcos + 2×fc[k]×cos(u×k×dx) + 4×fc[k+1]×cos(u×(k+1)×dx);
                       Fsin:= Fsin + 2×fs[k]×sin(u×k×dx) + 4×fs[k+1]×sin(u×(k+1)×dx);
                    end k;
                    Fc[u1] := Fcos×dx/3;
                    Fs[u1] := Fsin×dx/3;
                 end u;
              end x-integration;

              comment nu kendes Fc,Fs,Kc,Ks og u-integrationen kan
foretages (Simpson);
              uo:= entier(umax/0.1);
              us:= du/0.1;
              ymax:= -ln(lamin);
              ymin:= -ln(lamax);
              dy:= (ymax-ymin)/noofy;
              i:= 0;
              k1:= (Fc[0]×Kc[0]+Fs[0]×Ks[0])/(Kc[0]↑2+Ks[0]↑2);
              Kcs(¢<us¢, us);
              k2:= 4×(Fc[us]×Kc[us]+Fs[us]×Ks[us])/(Kc[us]↑2+Ks[us]↑2);
              k3:= 4×(Fs[us]×Kc[us]-Fc[us]×Ks[us])/(Kc[us]↑2+Ks[us]↑2);
              Kcs(¢<uo¢, uo);
              k4:= (Fc[uo]×Kc[uo]+Fs[uo]×Ks[uo])/(Kc[uo]↑2+Ks[uo]↑2);
              k5:= (Fs[uo]×Kc[uo]-Fc[uo]×Ks[uo])/(Kc[uo]↑2+Ks[uo]↑2);
              for y:=ymin step dy until ymax do
              begin
                 gy:= k1+k2×cos(y×du)+k3×sin(y×du)+k4×cos(y×umax)+k5×sin(y×umax);
                 for k:=2×us step 2×us until uo-2 do
                 begin
                    K:= k;
                    S:= 2;
OM:                 Fcu:= Fc[K];
                    Fsu:= Fs[K];
                    Kcs(¢<K ¢, K);
                    Kcu:= Kc[K];
                    Ksu:= Ks[K];
                    nævner:= Kcu↑2 + Ksu↑2;
                    u:= K×0.1;
                    gy:= gy+S×(Fcu×Kcu+Fsu×Ksu)/nævner×cos(y×u)
                        +S×(Fsu×Kcu-Fcu×Ksu)/nævner×sin(y×u);
                    K:=K + us;
                    S:= 4;
                    if K= k+us then goto OM;
                 end k;
                 g[i]:= gy;
                 i:= i+1;
              end y og u-integration;
           end integrationer;

           comment nu udskrives og plottes resultaterne;
           select(33);
           writecr;
           writecr;
           writetext(¢<dx,xmax,du,umax,lamax,lamin,noofy: ¢);
           writecr;
           write(¢-dddd.dddd¢, dx,xmax,du,umax,lamax,lamin,noofy);
           writecr;
           writecr;
           writetext(¢<y,lambda,g(exp(-y)): ¢);
           for i:=0 step 1 until noofy do
           begin
              y:= ymin + i×dy;
              lamda:= exp(-y);
              writecr;
```

```
   write(‹-ddd.dd›, y);
   write(‹-dddd.dddd›, lamda);
   writetext(‹<    ›);
   write(‹-ddd.d10+d›, g[i]);
end i;
gmax:= g[0];
for i:= 1 step 1 until noofy do if gmax<g[i] then gmax:= g[i];
writecr;
writecr;
writetext(‹<100: ›);
write(‹ddd10d›, gmax);
for j:=0 step 1 until noofy do g[j]:= g[j]/gmax×100;
select(17);
writecr;
writetext(‹<gør Calcomp klar til ny plotning›);
j:= read integer;
deltax:= 0.01×(ymax-ymin)/5;
deltay:= 0.01×100/5;
comment y-akses længde = 9 tommer;
xxx:= ymin;  yyy:= 0;
for i:= 10 step 10 until 140 do
begin
   plotline(ymin,i-10,ymin,i);
   plotline(ymin,i,ymin+10×deltax,i);
   if i=50 then
   begin
      x:= ymin-36×deltax;
      x:= plotchar(5,x,50,0.175);
      plotchar(16,x,50,0.175)
   end;
   if i=100 then
   begin
      x:= ymin-54×deltax;
      x:= plotchar(1,x,100,0.175);
      x:= plotchar(16,x,100,0.175);
      plotchar(16,x,100,0.175);
   end i=100;
end i;
for i:= -10 step -10 until -40 do
begin
   plotline(ymin,i+10,ymin,i);
   plotline(ymin,i,ymin+10×deltax,i)
end;
plotline(ymin,0,entier(ymin+1),0);
plotline(entier(ymin+1),0,entier(ymin+1),-2);
i:= 1;
for i:=i+1 while entier(ymin+i)≤ymax do
begin
   y:= entier(ymin+i);
   plotline(y-1,0,y,0);
   plotline(y,0,y,-2)
end;
plotline(entier(ymin+i-1),0,ymax,0);
plotline(ymax,-40,ymax-10×deltax,-40);
for i:=-30 step 10 until 140 do
begin
   plotline(ymax,i-10,ymax,i);
   plotline(ymax,i,ymax-10×deltax,i)
end;
for i:=1 step 1 until noofy do plotline(ymin+(i-1)×dy,g[i-1],ymin+i×dy,g[i])
i:=10;  lstep:= 1;
for i:=i×10 while entier(i×lamax)=0 do lstep:= 1/i/10;
i:= 0;  y:= ymin;  lamda:= lamax;
for i:= i+1 while y<ymax do
```

```
         begin
            for k:= -1 step 1 until 4 do
            if abs(lamda-10↑(-k))<10-7 then
            begin
               if k≤0 then
               begin
                  x:=plotchar(1,y-k×8×deltax,-49,0.15);
                  for i:=k step 1 until -1 do x:=plotchar(16,x,-49,0.15);
               end
               else
               begin
                  x:= y-k×8×deltax;
                  x:= plotchar(16,x,-49,0.03);
                  for i:=2 step 1 until k do x:= plotchar(16,x,-49,0.15);
                  plotchar(1,x,-49,0.15)
               end
            end;
            lamda:= lamda - lstep;
            if lamda<10-7 then
            begin
               lamda:= lamda+lstep-lstep/10;
               lstep:= lstep/10
            end;
            y:= -ln(lamda);
            if y>ymax then
            begin
               plotline(-ln(lamda+lstep),-40,ymax,-40);
               goto l1
            end;
            plotline(-ln(lamda+lstep),-40,y,-40);
            plotline(y,-40,y,-42);
l1:      end lamdaakse 1;
         i:= 10;
         lstep:= 1;
         for i:=i×10 while entier(i×lamax)=0 do lstep:=1/i/10;
         i:= 0;
         y:= ymin;
         lamda:= lamax;
         for i:=i+1 while y<ymax do
         begin
            lamda:= lamda - lstep;
            if lamda<10-7 then
            begin
               lamda:= lamda+lstep-lstep/10;
               lstep:= lstep/10
            end;
            y:= -ln(lamda);
            if y>ymax then
            begin
               plotline(-ln(lamda+lstep),140,ymax,140);
               goto l2
            end;
            plotline(-ln(lamda+lstep),140,y,140);
            plotline(y,140,y,138);
l2:      end lamdaakse 2;
      end næstyderste blok;
      nykurve:= false;
      goto DATA;
   end yderste blok
end;
t<
```