DISTORTION OF FORM ON COMPUTER

SPECIAL PROJECT REPORT

I. D. C. Library
I. I. T. Bombay.

Ravindra Rajhans

DISTORTION OF FORM ON COMPUTER (PDP 11)

BY
RAVINDRA RAJHANS

GUIDE
PROF.U.A.ATHAVANKAR

I. D. C. Library
I. I. T. Bombay.

INDUSTERRIAL DESIGN CENTRE
INDIAN INSTITUTE OF TECHNOLOGY
BOMBAY 400076

APPROVAL SHEET

Special project entitled 'DISTORTION OF FORM ON COMPUTER (PDP 11)' by Ravindra Rajhans is approved in partial fulfilment of the requirements for Master degree in Industrial Design.

Guide

Prof. U.A. Athavankar

ACKNOWLEDGEMENTS

I sincerely thank my guide Professor U.A. Athavankar and Mr. N.Sadhu for their invaluable guidence at every stage of this seminar.

I am also grateful to my classmates for their healthy suggestions and cribs.

IIT-Bombay RAVINDRA RAJHANS

May, 1987

INDEX

CONTENTS

INTRODUCTION

It is a very difficult to define form. Generally it can be explained or described the shape of an object in the simple words of common speech. But this ends by defining it in precise language of mathematics and one method tends to follow the other in strict scientific order and historical continuity.

Mathematical definitions are too strict and rigid for common use, but their rigour is combined with all but endless freedom.

The precise definition of an 'Ellipse' introduces all the ellipses in the world. By this controlled and regulated freedom any body can reach through mathematical analysis to synthesis.

The details in which the figure differes from its mathematical prototype are more important and interesting than the features in which it agrees. Even the peculiar aesthetic pleasure with which common man regard a living object somehow bound up with the departure from mathematical regularity which it manifests as a peculiar attribute of life. In the morphology of living things the use of mathematic's methods and symbols has made slow progress.

According to 'Louis Sullivan' form followes function' But form is not dependent only on function, it also responds forces acting on it may be mathematical, environmental, historical. In nature there is indefinite variety, again many times keeping some law/rule. Slight complexity becomes far beyond of man's mental capacity to analyse. The change in living object can be called its transition and for man made objects this variation can be obtained by distroying the prototypical form of an object.

MATHEMATICS

* Take a cartetion coordinate network
* Any figure can be treated as function of 'X' and 'Y' for e.g. fish, crab, human skull, a cup, jug, aeroplane or a transistor.
* Suppose by the combined action of appropriate forces any material form can be transformed into any other. Just like out of 'shapeless' mass of clay the potter or sculpturer models his artistic products.
* Take an example of circle

  Circle can be transformed into an ellipse when coordinate grid distroyed unidirectionally.

  On paper it can be done by writting equation of circle and changing it to equation of ellipse i.e.

  Equation of circle is $x^2 + y^2 = a^2$ and it becomes equation of ellipse $x^2 + c^2y^2 = a^2$ i.e 'y' gets transformed to 'cy'

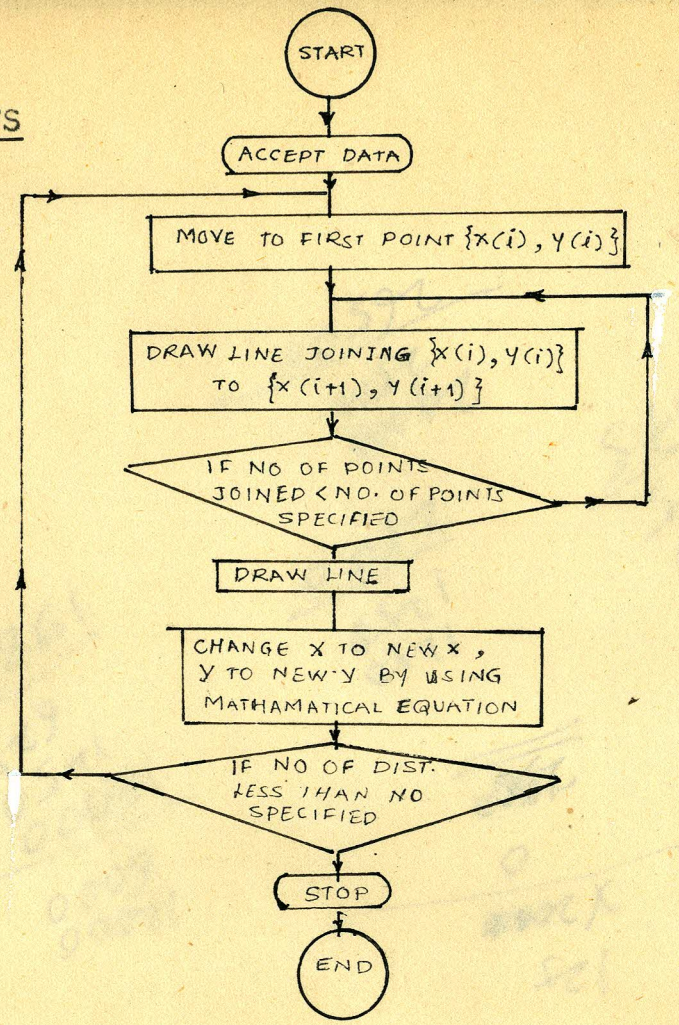
  When 'x' or 'y' or both multiplied and /or divided by sine, cosine, tangent of some angle, it gives simple shear of the grid.

* This can be a complex transformation i.e. equation of line $y = mx + c$ converting to logarathemic spiral $\theta - A \log (r) = c$

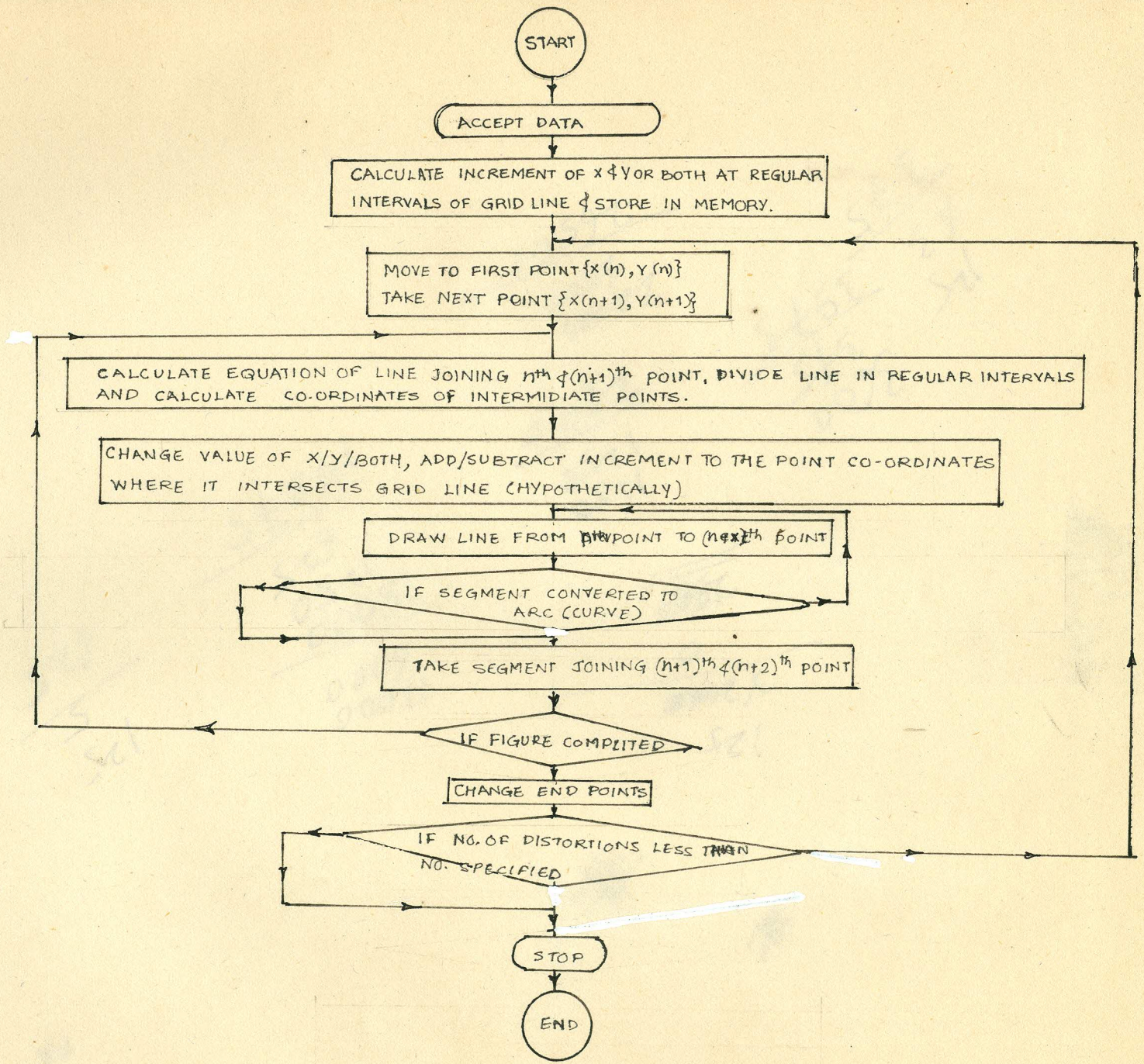
* For doing these mathematical calculations computer can be used because

Human brain	Computer
* It is creative	* It is a machine
* It can do mechanical work upto a limit	* It can work mechanically for unlimited time (depends on capacity of the computer)
* Fairly accurate for judgement but not fast in calculations	* It can not judge but very fast in calculating equations for several times.
* Accuracy reduces with fatigue	* Accuracy remains same
* Ideal for judgemental work, decision making psychological and creative work.	* Ideal for making large number of alternatives calculations and for studying complicated shapes.

FLOW CHARTS



GROUP A - Distortion by end point movement



GROUP B - Distortion of figure to specified radius

THE PROGRAMME

* Execute the programme 'CHITRA'
* Draw a figure having straight lines and maximum 100 end points on a graph paper having scale.
* Calculate end point co-ordinates in sequence starting from any point of the figure.
* Feed number of points to be given.
* Feed number of subroutines you want to run
* Feed value of radius for vertical line distortion(250 to 2000 units). Feed value of radius for horizontal line (150 to 1000 units)
* Feed co-ordinates of points in sequence
* Programme will run after completion of data entry
* After drawing of figure on screen programme will hault and will go in pause mode
* Trace the figure or shoot it and then press return key to come out of pause mode.
* Programme will stop automatically after distorting the figure by number of times fed previously.
* Drafting package can be used for sketching the figures obtained

SCREEN Vs POSITION OF FIGURE

Basically figure is imposed on a grid and that grid is distroyed by using different mathematical equations, so even though basic distortion of a grid is same, position of figure with respect to that grid plays vital role. If it is placed in a corner or in centre or somewhere else distroyed form will be slightly different. This gives advantage for obtaining many more distroyed alternatives just by changing position of figure with respect to screen.

APPLICATIONS

* This method can be used for studying transition of living objects with time or finding out relation between forms of living objects of same species for e.g. studying form transition in fishes.

* To get different forms of a letter which can be used for designing different letter styles.

* In animation many times pictures are in slightly transformed series so by feeding a basic picture many more transformed alternatives can be obtained which can be used as a basic form for finish art work.

* In product design, many times object is recognised as a cup by common man when he feels some cupness in its form. This programming method can be used for getting many more alternative by feeding a prototypical form of object. Again these alternatives can be used for studying when prototypicalness of an object destroys after destroying features of that prototypical form of an object. Many more uses can be explored by using different mathematical equations and methods.

LIMITATIONS OF PDP 11

PDP 11 is a micro computer terminal. It has a graphics package but it is of very primary level, just drawing a two dimensional figure having straight lines, curves of definite radius, shadding figure by 4 shades etc.

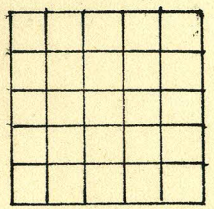
Again PDP11 has very small memory capacity which doesn't allow to process a figure beyond 100 points.

Methods can be explored on some different computer having bigger memory capacity keeping the principles of distortion same.

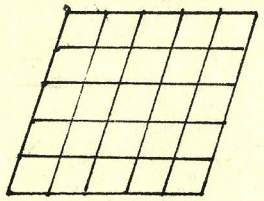
FUTURE SCOPE - Here a standard programme had been developed to process a figure having 100 points and these points joined by straight lines having continuity.

Basis of distortion is as follows

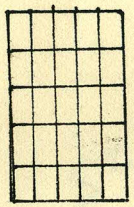
GROUP A



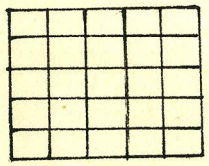
Basic grid



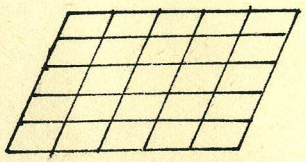
Simple shear



Compression of horizontal dim.^{n}

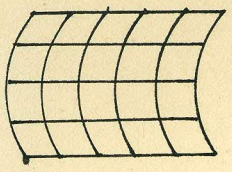


Elongation of horizontal dimension

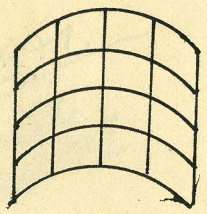


Elongation or/and compression of horizontal or vertical dimension combined with shear.

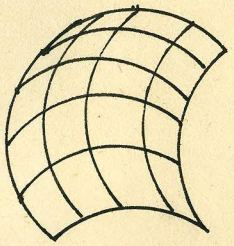
GROUP B



Distortion of vertical line to specified are



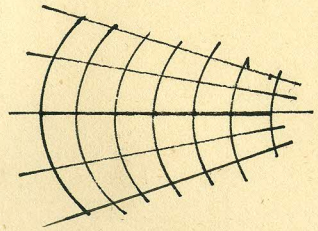
Distortation of horizontal line to specified arc

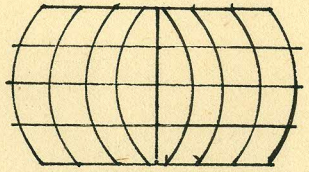


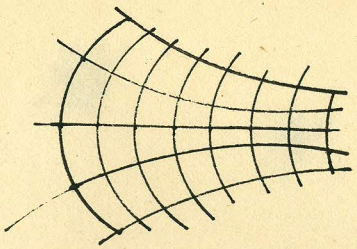
Distortation of horizontal and vertical line to specified arc

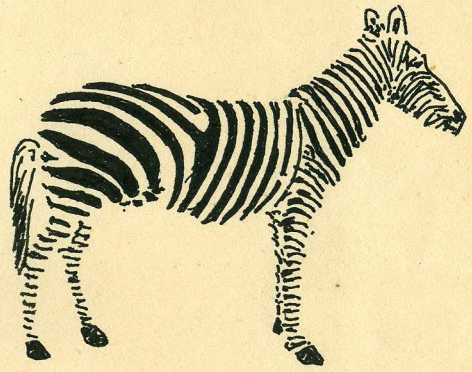
Combining Group A and Group B for getting many more combinations

This method can be explored to get many more alternatives by destroying point location w.r.t x , y, grid by different mathematical equations like logarithmic spiral varying arcs changing from left to right or side to centre etc. or for studying and distorting patterns.





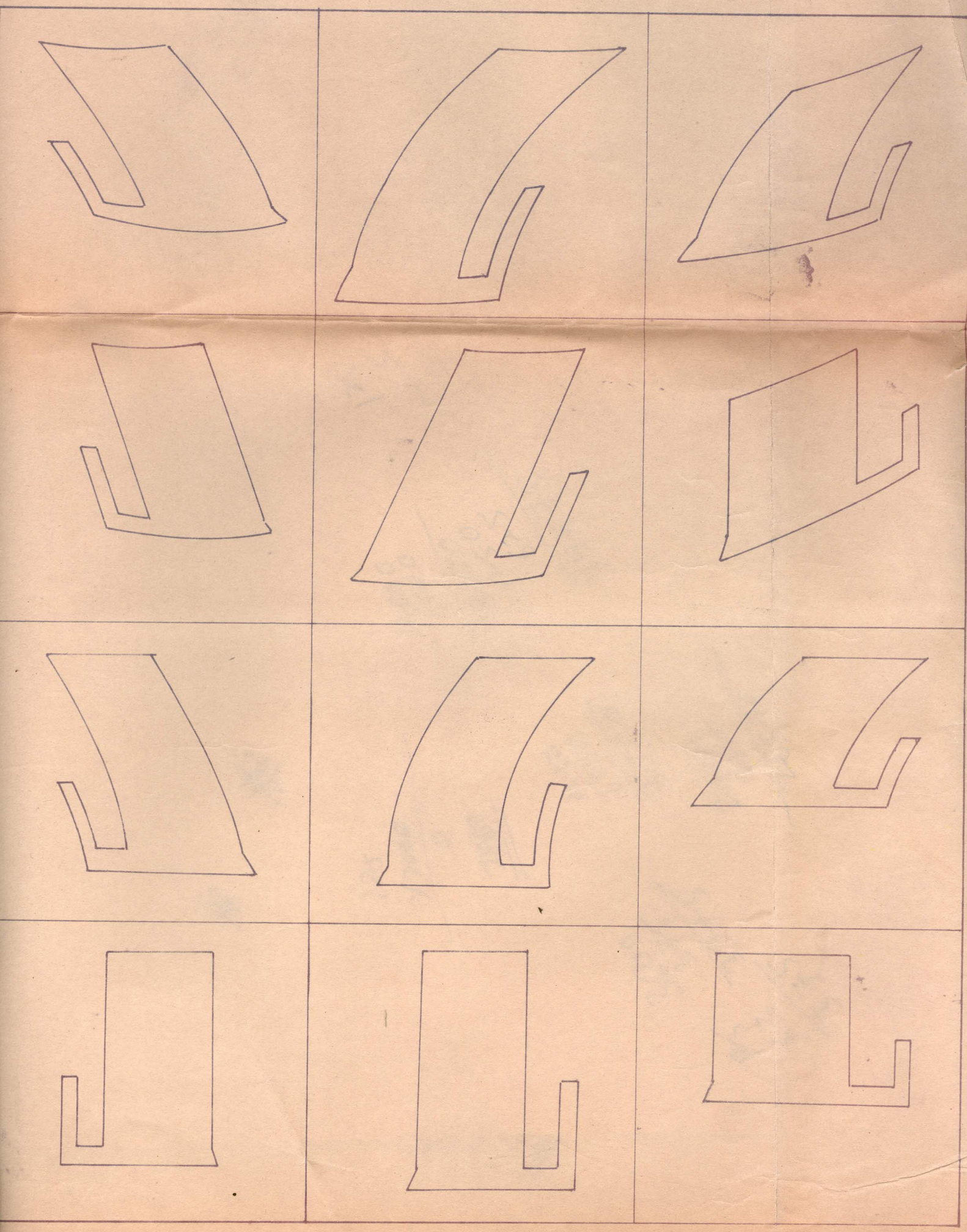


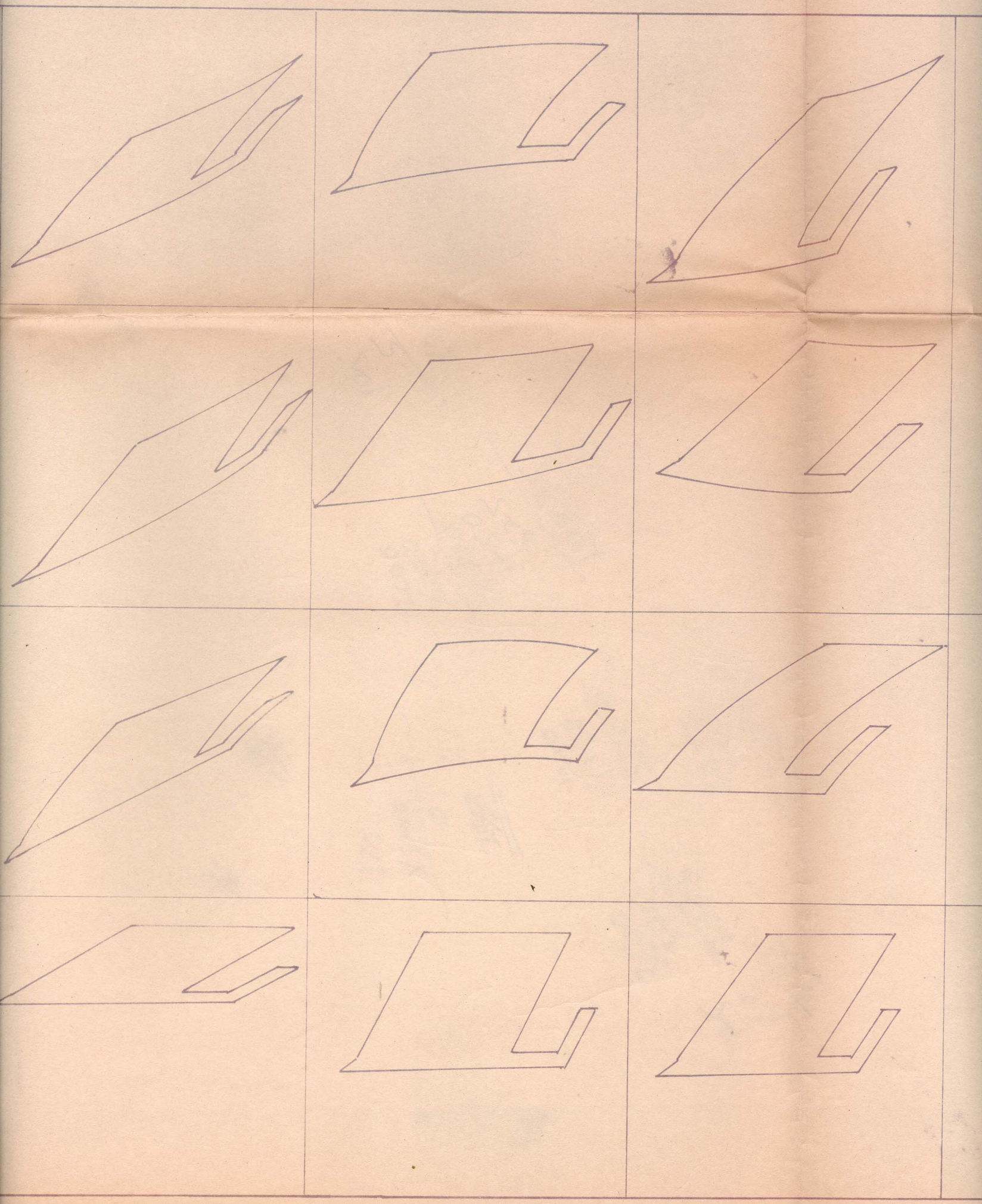


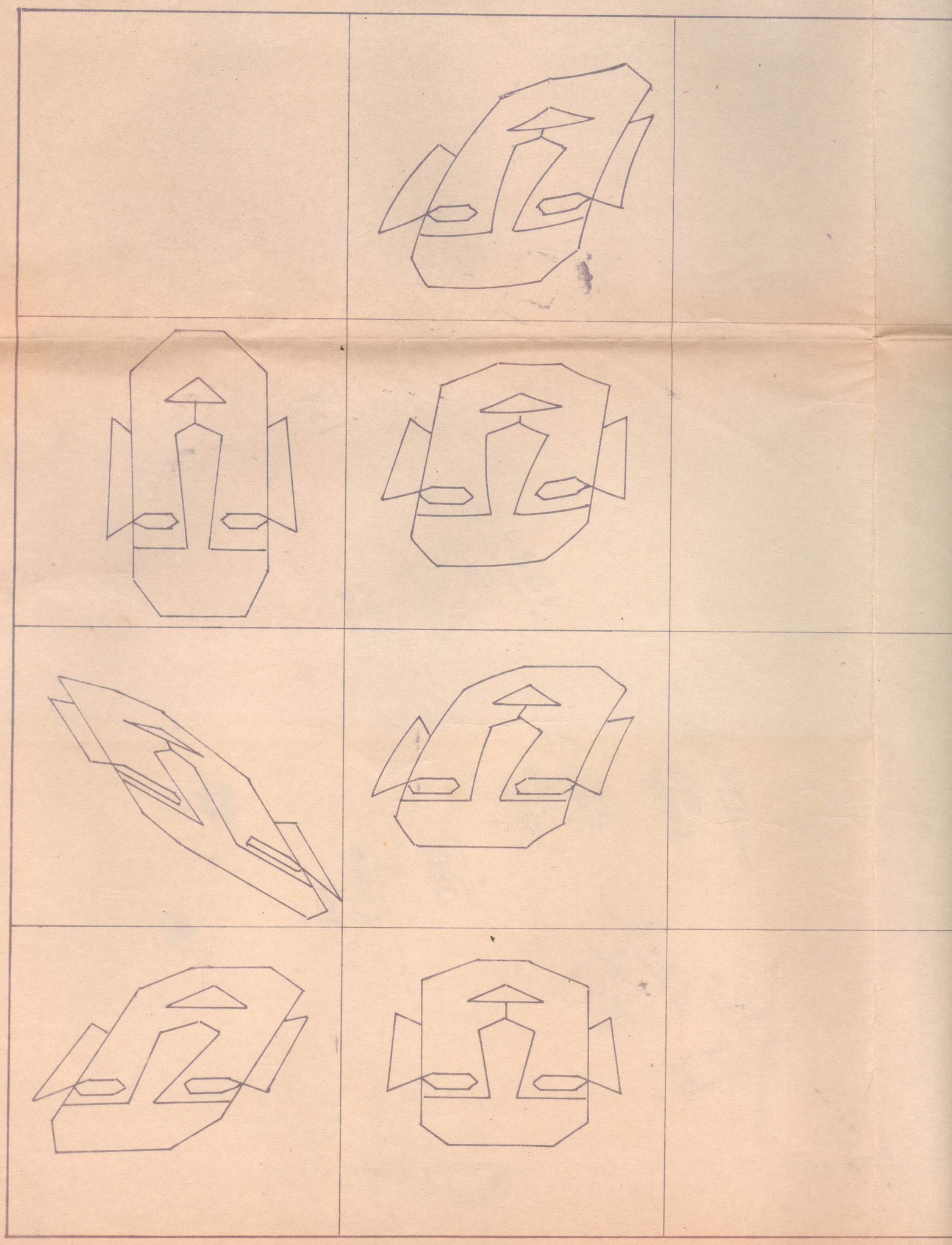
I. D. C. Library
I. I. T. Bombay.

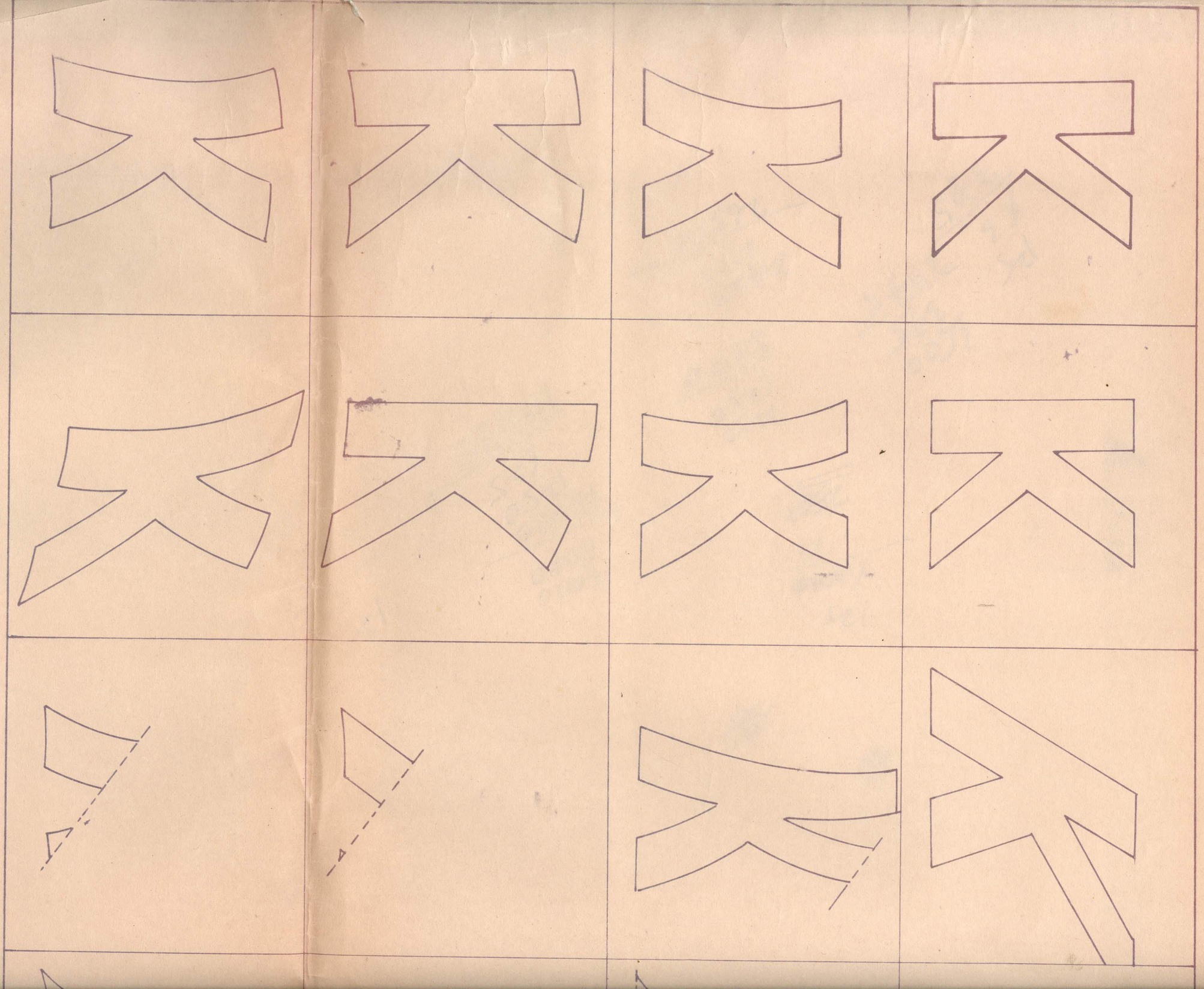
REFERENCES

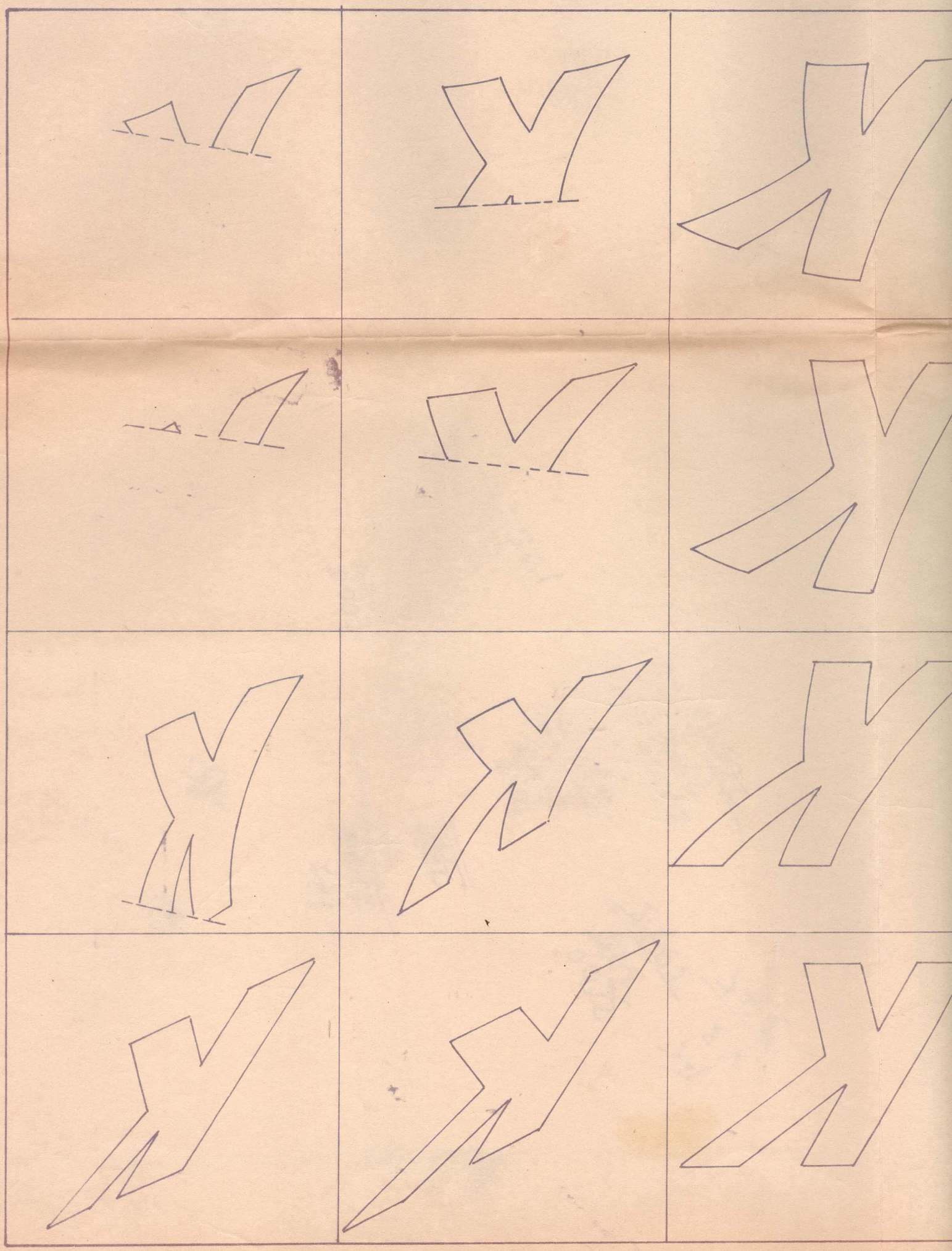
* D'Arcy wentworth Thompson, 'Growth and Form', Cambridge, University Press 1968, pp. 1026 - 1095.











FOR DETAILS AND PROGRMMING FUNDAMENTALS REFER FORTRAN IV AND PDP 11 GRAPHICS PACKAGE MANUAL.

Programme to find out best praportion of a figure.

(x, y), (a, b) = co-ordinates of points

```
program KD
BY RAVINDRA RAJHANS
dimension x(50),y(50),a(50),b(50)
common/blk/a,b
type 1
format(' ENTER NUMBER OF POINTS ')
accept 2,no
format(i4)
do 10 i=1,no
accept 4,a(i)
format(f7.2)
x(i)=a(i)
accept 4,b(i)
y(i)=b(i)
if(i.lt.no) type 6
format(' ENTER NEXT VALUE ')
continue
type 7
format(' DATA ENTRY COMPLETED ')
z=0.5
call initgr(5)
call clrscr
call clrtxt
do 50 i=1,15
call vart1(x,y,no,z)
call drawn(x,y,no)
z=z+0.1
pause 'trace'
call clrscr
call clrtxt
continue
zz=0.5
call clrscr
call clrtxt
do 80 i=1,20
call vart2(x,y,no,zz)
call drawn(x,y,no)
zz=zz+0.1
pause 'trace'
call clrscr
call clrtxt
continue
```

```
      call clrscr
      call clrtxt
      stop
      end

      subroutine vart1(x,y,no,z)
      dimension x(50),y(50),a(50),b(50)
      common/blk/a,b
      do 60 i=1,no
      x(i)=a(i)
      y(i)=b(i)*z
0     continue
      return
      end

      subroutine vart2(x,y,no,zz)
      dimension x(50),y(50),a(50),b(50)
      common/blk/a,b
      do 70 i=1,no
      x(i)=a(i)*zz
      y(i)=b(i)
0     continue
      return
      end

      subroutine drawn(x,y,no)
      dimension x(50),y(50),a(50),b(50)
      common/blk/a,b
      call initgr(5)
      call swindo(0.,0.,400.,250.)
      call box(0.,0.,400.,250.)
      call slnpat(1, )
      i=1
      call move(x(i),y(i))
0     i=i+1
      call line(x(i),y(i))
      if(i.eq.no) go to 25
      go to 30
5     call line(x(1),y(1))
      return
      end
```

keeping x co-opdinate same
change y co-ordinate

keeping y co-ordinate same
change x co-ordinate

Joining points given in (x, y) co-ordinate system and order

Programme to distort the figure to the way specified by mathematical equation.

```
program CHITRA
BY RAVINDRA RAJHANS
APRIL 1987
dimension x(150),y(150),a(150),b(150),g(500)
dimension p(500),q(500),c(450),d(450),xincr(450)
type 1
format(' ENTER NUMBER OF POINTS ')
accept 2,no
format(i5)
type 3
format(' ENTER VALUE OF R FOR VERT.LINE ')
accept 4,r
format(f8.2)
if(r.le.250.) go to 6
if(r.ge.1000.) go to 7
go to 20
type 8
format(' R SHOULD BE GREATER THAN 250. ')
go to 9
type 10
format(' R SHOULD BE LESS THAN 1000. ')
type 11
format(' MODIFY VALUE OF RADIUS ')
accept 4,r
go to 5
type 32
format(' ENTER VALUE OF R FOR HORZ.LINE ')
accept 33,rr
format(f8.2)
if(rr.le.400.)go to 36
if(rr.ge.2000.) go to 37
go to 35
type 38
format(' RR SHOULD BE GREATER THAN 400. ')
go to 39
type 42
format(' RR SHOULD BE LESS THAN 2000. ')
type 43
format(' MODIFY VALUE OF RADIUS ')
accept 33,rr
go to 34
type 12
format(' ENTER COORDINATES IN SEQUENCE ')
do 100 i=1,no
```

r = Radius value for vertical line distortion

rr = Radius value for horizontal line distortion.

```
accept 13,1)
format(f7
x(i)=a(i),b(i)
accept 13
y(i)=b(i
type 14'              ')
format(.no) type 15
if(i.lt ' ENTER NEXT VALUE ')
format(
continue
type 16 ' DATA ENTRY COMPLITED ')
format(awl(a,b,x,y,no)
call dr n=1,10
do 102 .1) call dist1(a,b,x,y,no)
if(n.eq.2) call dist2(a,b,x,y,no)
if(n.eq.3) call dist3(a,b,x,y,no)
if(n.eq.4) call dist4(a,b,x,y,no)
if(n.eq.5) call dist5(a,b,x,y,no)
if(n .eq.6) call dist6(a,b,x,y,no)
if(n.eq.7) call dist7(a,b,x,y,no)
if(n.eq.8) call dist8(a,b,x,y,no)
if(n .eq.9) call dist9(a,b,x,y,no)
if(n .eq.10) call dist10(a,b,x,y,no)
if(n 02 k=1,4
do 1k.eq.1) call draw1(a,b,x,y,no)
if(k.eq.2) call draw2(x,y,p,q,u,v,s,t,kk,r,c,d,g,xincr)
if((k.eq.3) call draw3(x,y,p,q,u,v,s,t,kk,rr,c,d,g,xincr)
if (k.eq.4) call draw4(x,y,p,q,u,v,s,t,kk,r,rr,c,d,g,xincr)
ifontinue
crause 'end'
p:all clrscr
(call clrtxt
stop
end
```

Calling subroutines which specifies the mathematical equation by which end point co-ordinates value of a segment of a figure to be distorted.

Calling subroutine which distorts (x,y) co-ordinate frame horizontaly, verticly or both way to specified radius.

```
subroutine draw1(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
call initgr(5)
call swindo(0.,0.,800.,500.)
call clrscr
call clrtxt
call box(0.,0.,800.,500.)
call slnpat(1, )
i=1
call move(x(i),y(i))
i=i+1
call line(x(i),y(i))
if(i.eq.no) go to 31
go to 30
call line(x(1),y(1))
pause 'trace'
call clrscr
call clrtxt
return
end
```

Conecting the specified points in specified sequance to draw a figure.

```
subroutine draw2(x,y,p,q,u,v,s,t,kk,r,c,d,g,xincr)
dimension x(150),y(150),p(500),q(500),g(500)
dimension c(450),d(450),xincr(450)
call initgr(5)
call swindo(0.,-250.,800.,250.)
call clrscr
call clrtxt
call box(0.,-250.,800.,250.)
call slnpat(1, )
mz=1
s=x(mz)
```

```
mz=mz+1
call coord(p,q,u,v,s,t,kk)
call shixx(c,d,p,q,r,kk,g,xincr)
call sket(p,q)
if(mz.eq.no) go to 201
go to 200
s=x(no)
u=x(1)
t=y(no)-250.
v=y(1)-250.
call coord(p,q,u,v,s,t,kk)
call shixx(c,d,p,q,r,kk,g,xincr)
call sket(p,q)
pause 'trace'
call clrscr
call clrtxt
return
end
```

Distorting the (x, y) frame of referance to specified radius horizontaly.

(s, t), (u, v) co-ordinates of end points of a specified segment of the figure.

```
subroutine draw3(x,y,p,q,u,v,s,t,kk,rr,c,d,g,xincr)
dimension x(150),y(150),p(500),q(500),g(500)
dimension c(450),d(450),xincr(450)
call initgr(5)
call swindo(-400.,0.,400.,500.)
call clrscr
call clrtxt
call box(-400.,0.,400.,500.)
call slnpat(1, )
mz=1
s=x(mz)-400.
u=x(mz+1)-400.
t=y(mz)
v=y(mz+1)
mz=mz+1
call coord(p,q,u,v,s,t,kk)
call shiyy(c,d,p,q,rr,kk,g,xincr)
call sket(p,q)
if(mz.eq.no) go to 211
go to 210
s=x(no)-400.
u=x(1)-400.
t=y(no)
v=y(1)
call coord(p,q,u,v,s,t,kk)
call shiyy(c,d,p,q,rr,kk,g,xincr)
call sket(p,q)
call move(-400.,0.)
pause 'trace'
call clrscr
call clrtxt
return
end
```

```
subroutine draw4(x,y,p,q,u,v,s,t,kk,r,rr,c,d,g,xincr)
dimension x(150),y(150),p(500),q(500),g(500)
dimension c(450),d(450),xincr(450)
call initgr(5)
call swindo(-400.,-250.,400.,250.)
call clrscr
call clrtxt
call box(-400.,-250.,400.,250.)
call slnpat(1, )
mz=1
s=x(mz)-400.
u=x(mz+1)-400.
t=y(mz)-250.
v=y(mz+1)-250.
mz=mz+1
call coord(p,q,u,v,s,t,kk)
```

Distorting the (x, y) frame of referance to specified radius vertically.

```
call shixx(c,d,p,q,r,kk,g,xincr)
call shiyy(c,d,p,q,rr,kk,g,xincr)
call sket(p,q)
if(mz.eq.no) go to 221
go to 220
s=x(no)-400.
u=x(1)-400.
t=y(no)-250.
v=y(1)-250.
call coord(p,q,u,v,s,t,kk)
call shixx(c,d,p,q,r,kk,g,xincr)
call shiyy(c,d,p,q,rr,kk,g,xincr)
call sket(p,q)
call move(-400.,-250.)
pause 'trace'
call clrscr
call clrtxt
return
end
```

Distorting the (x,y) frame of referance horizontaly, as well as verticaly to specified radius.

```
subroutine coord(p,q,u,v,s,t,kk)
dimension p(500),q(500)
if(s.eq.u) go to 251
if(t.eq.v) go to 252
k=1
p(k)=s
uu=(v-t)/(u-s)
vv=(u*t-s*v)/(u-s)
if(s.lt.u) incr=1.
if(s.gt.u) incr=-1.
q(k)=((uu*p(k)))+vv
k=k+1
p(k)=p(k-1)+incr
kk=k-1
if((abs(p(k)-u)).lt.0.5) go to 255
go to 253
k=1
p(k)=s
q(k)=t
if(t.lt.v) incr=1.
if(t.gt.v) incr=-1.
k=k+1
q(k)=q(k-1)+incr
p(k)=s
kk=k
if((abs(q(k)-v)).lt.0.5) go to 255
go to 254
k=1
p(k)=s
q(k)=t
if(s.lt.u) incr=1.
if(s.gt.u) incr=-1.
k=k+1
p(k)=p(k-1)+incr
q(k)=t
kk=k
if((abs(p(k)-u)).lt.0.5) go to 255
go to 256
k=1
return
end
```

Calculating equation of line joining two points of specified segments, breaking segment at equal interval, calculating values of co-ordinates of all those points.

```
subroutine shixx(c,d,p,q,r,kk,g)
dimension c(450),d(450),p(500),q(500),g(500),xincr(450)
z=r**2-250.**2
xint=abs(sqrt(abs(z)))
d(0)=-1.
i=0
do 125 i=1,251
```

```
zz=((r**2)-(d(i)**2))
zz=abs(zz)
c(i)=abs(sqrt(zz))
xincr(i)=c(i)-xint
5 continue
do 126 lo=1,(kk-1)
g(lo)=p(lo)
6 continue
do 127 l=1,(kk-1)
w=abs(g(l))
do 127 n=1,251
if(abs(w-d(n)).le.0.5) g(l)=g(l)+xincr(n)
if(w.eq.0.0) go to 127
if(abs(g(l)-p(l)).eq.0.0) go to 127
diff=abs(g(l)-p(l)-xincr(n)-xincr(n-1))
if(diff.lt.0.001) g(l)=g(l)-xincr(n)
7 continue
do 128 lo=1,(kk-1)
p(lo)=g(lo)
8 continue
return
end
```

Changing x co-ordinate of points wh
y changes from one point to another
so that figure's x co-ordinate will
change to distort figure horizontally
with a hypothetical distorted fram
of referance.

```
subroutine shiyy(c,d,p,q,rr,kk,g,xincr)
dimension c(450),d(450),p(500),q(500),g(500),xincr(450)
z=rr**2-400.**2
yint=abs(sqrt(abs(z)))
c(0)=-1.
i=0
do 175 i=1,401
c(i)=c(i-1)+1.
zz=((rr**2)-(c(i)**2))
zz=abs(zz)
d(i)=abs(sqrt(zz))
xincr(i)=d(i)-yint
5 continue
do 176 lo=1,(kk-1)
g(lo)=q(lo)
6 continue
do 177 l=1,(kk-1)
w=abs(p(l))
do 177 n=1,401
if(abs(w-c(n)).le.0.5) g(l)=g(l)+xincr(n)
if(w.eq.0.0) go to 177
if(abs(g(l)-q(l)).eq.0.0) go to 177
diff=abs(g(l)-q(l)-xincr(n)-xincr(n-1))
if(diff.lt.0.001) g(l) =g(l)-xincr(n)
7 continue
do 178 lo=1,(kk-1)
q(lo)=g(lo)
8 continue
return
end
```

Changing y co-ordinate of points whe
x changes from one point to anothe
so that figures y co-ordinate will
change to distort figure vertically
with a hypothetical distorted fram
of referance.

```
subroutine sket(p,q)
dimension p(500),q(500)
jz=1
call move(p(jz),q(jz))
jz=jz+1
call line(p(jz),q(jz))
if(jz.eq.kk) go to 61
go to 60
return
end
```

Joining two end points
specified in order.

```
subroutine distl(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 300 i=1,no
```

```
y(i)=b(i)
continue
return
end


subroutine dist2(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 301 i=1,no
x(i)=a(i)
y(i)=b(i)+a(i)*sin(3.14/3)/cos(3.14/3)
continue
return
end


subroutine dist3(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 302 i=1,no
x(i)=2.*a(i)
y(i)=0.8*b(i)
continue
return
end


subroutine dist4(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 303 i=1,no
x(i)=a(i)+50.
y(i)=b(i)-40.
continue
return
end


subroutine dist5(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 304 i=1,no
x(i)=(a(i)*(sin(a(i)*3.14/180)))/10.
y(i)=(b(i)*(cos(b(i)*3.14/180)))/10.
continue
return
end


subroutine dist6(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 305 i=1,no
x(i)=0.7*a(i)
y(i)=1.2*b(i)
continue
return
end


subroutine dist7(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 306 i=1,no
x(i)=1.2*a(i)+b(i)*sin(0.5235)/cos(0.5235)
y(i)=0.7*b(i)+a(i)*sin(0.5235)/cos(0.5235)
continue
return
end


subroutine dist8(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 307 i=1,n0
x(i)=0.8*a(i)-b(i)*sin(1.05)
y(i)=1.4*b(i)-a(i)*sin(1.05)
```

Distorting (transfering) (x,y) co-ordinates by specified Mathematical equations.

```
return
end


subroutine dist9(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 308 i=1,no
x(i)=0.8*abs(sqrt(a(i)*b(i)))
y(i)=1.2*b(i)
continue
return
end


subroutine dist10(a,b,x,y,no)
dimension x(150),y(150),a(150),b(150)
do 309 i=1,no
x(i)=1.5*a(i)
y(i)=abs(sqrt(a(i)*b(i)))
continue
return
end
```

```
program CHITRA
BY RAVINDRA RAJHANS
APRIL 1987
dimension x(50),y(50),a(50),b(50),g(200)
dimension p(200),q(200),c(210),d(210),xincr(210)
common/blk1/a,b/blk2/p,q/blk3/c,d,xincr
open (unit=11,name='DAT.DAT',type='OLD')
read (11,2) no
format(1x,i5)
type 3
format(' ENTER VALUE OF R FOR VERT.LINE ')
accept 4,r
format(f8.2)
type 32
format(' ENTER VALUE OF R FOR HORZ.LINE ')
accept 33,rr
format(f8.2)
type 5
format(' ENTER VALUE OF SUBROUTINE ')
accept 6,mq
format(i3)
type 12
format(' ENTER COORDINATES IN SEQUENCE ')
do 100 i=1,no
read (11,13) a(i)
format(1x,f7.1)
x(i)=a(i)
read (11,13) b(i)
y(i)=b(i)
continue
close (unit=11)
type 16
format(' DATA ENTRY COMPLITED ')
if(no.eq.9) go to 444
do 104 nn=1,4
if(nn.eq.1) call draw1(x,y,no)
if(nn.eq.2) call draw2(x,y,u,v,s,t,kk,r,g,no)
if(nn.eq.3) call draw3(x,y,u,v,s,t,kk,rr,g,no)
if(nn.eq.4) call draw4(x,y,u,v,s,t,kk,r,rr,g,no)
continue
do 102 nn=1,mq
if(nn.eq.1) call dist1(x,y,no)
if(nn.eq.2) call dist2(x,y,no)
```

```
if(nn.eq.4) call dist4(x,y,no)
if(nn.eq.5) call dist5(x,y,no)
if(nn.eq.6) call dist6(x,y,no)
if(nn.eq.7) call dist7(x,y,no)
if(nn.eq.8) call dist8(x,y,no)
do 102 kn=1,4
if(kn.eq.1) call draw1(x,y,no)
if(kn.eq.2) call draw2(x,y,u,v,s,t,kk,r,g,no)
if(kn.eq.3) call draw3(x,y,u,v,s,t,kk,rr,g,no)
if(kn.eq.4) call draw4(x,y,u,v,s,t,kk,r,rr,g,no)
continue
pause 'end'
call clrscr
call clrtxt
stop
end

subroutine draw1(x,y,no)
dimension x(50),y(50),a(50),b(50)
common/blk1/a,b
call initgr(5)
call swindo(0.,0.,400.,250.)
call clrscr
call clrtxt
call box(0.,0.,400.,250.)
call slnpat(1, )
i=1
call move(x(i),y(i))
i=i+1
call line(x(i),y(i))
if(i.eq.no) go to 31
go to 30
call line(x(1),y(1))
pause 'trace'
call clrscr
call clrtxt
return
end

subroutine draw2(x,y,u,v,s,t,kk,r,g,no)
dimension x(50),y(50),p(200),q(200),g(200)
dimension c(210),d(210),xincr(210)
common/blk2/p,q/blk3/c,d,xincr
call initgr(5)
call swindo(0.,-125.,400.,125.)
call clrscr
call clrtxt
call box(0.,-125.,400.,125.)
call slnpat(1, )
mz=1
s=x(mz)
u=x(mz+1)
t=y(mz)-125.
v=y(mz+1)-125.
mz=mz+1
call coord(u,v,s,t,kk)
call shixx(r,kk,g)
call sket(kk)
if(mz.eq.no) go to 201
go to 200
s=x(no)
u=x(1)
t=y(no)-125.
v=y(1)-125.
call coord(u,v,s,t,kk)
call shixx(r,kk,g)
call sket(kk)
pause 'trace'
call clrscr
call clrtxt
```

```
end

subroutine draw3(x,y,u,v,s,t,kk,rr,g,no)
dimension x(50),y(50),p(200),q(200),g(200)
dimension c(210),d(210),xincr(210)
common/blk2/p,q/blk3/c,d,xincr
call initgr(5)
call swindo(-200.,0.,200.,250.)
call clrscr
call clrtxt
call box(-200.,0.,200.,250.)
call slnpat(1, )
mz=1
s=x(mz)-200.
u=x(mz+1)-200.
t=y(mz)
v=y(mz+1)
mz=mz+1
call coord(u,v,s,t,kk)
call shiyy(rr,kk,g)
call sket(kk)
if(mz.eq.no) go to 211
go to 210
s=x(no)-200.
u=x(1)-200.
t=y(no)
v=y(1)
call coord(u,v,s,t,kk)
call shiyy(rr,kk,g)
call sket(kk)
call move(-200.,0.)
pause 'trace'
call clrscr
call clrtxt
return
end

subroutine draw4(x,y,u,v,s,t,kk,r,rr,g,no)
dimension x(50),y(50),p(200),q(200),g(200)
dimension c(210),d(210),xincr(210)
common/blk2/p,q/blk3/c,d,xincr
call initgr(5)
call swindo(-200.,-125.,200.,125.)
call clrscr
call clrtxt
call box(-200.,-125.,200.,125.)
call slnpat(1, )
mz=1
s=x(mz)-200.
u=x(mz+1)-200.
t=y(mz)-125.
v=y(mz+1)-125.
mz=mz+1
call coord(u,v,s,t,kk)
call shixx(r,kk,g)
call shiyy(rr,kk,g)
call sket(kk)
if(mz.eq.no) go to 221
go to 220
s=x(no)-200.
u=x(1)-200.
t=y(no)-125.
v=y(1)-125.
call coord(u,v,s,t,kk)
call shixx(r,kk,g)
call shiyy(rr,kk,g)
call sket(kk)
call move(-200.,-125.)
pause 'trace'
call clrscr
call clrtxt
```

```
end

subroutine coord(u,v,s,t,kk)
dimension p(200),q(200)
common/blk2/p,q
if(s.eq.u) go to 251
if(t.eq.v) go to 252
k=1
p(k)=s
uu=(v-t)/(u-s)
vv=(u*t-s*v)/(u-s)
if(s.lt.u) incr=1.
if(s.gt.u) incr=-1.
q(k)=((uu*p(k)))+vv
k=k+1
p(k)=p(k-1)+incr
kk=k-1
if((abs(p(k)-u)).lt.0.5) go to 255
go to 253
k=1
p(k)=s
q(k)=t
if(t.lt.v) incr=1.
if(t.gt.v) incr=-1.
k=k+1
q(k)=q(k-1)+incr
p(k)=s
kk=k
if((abs(q(k)-v)).lt.0.5) go to 255
go to 254
k=1
p(k)=s
q(k)=t
if(s.lt.u) incr=1.
if(s.gt.u) incr=-1.
k=k+1
p(k)=p(k-1)+incr
q(k)=t
kk=k
if((abs(p(k)-u)).lt.0.5) go to 255
go to 256
k=1
return
end

subroutine shixx(r,kk,g)
dimension c(210),d(210),p(200),q(200),g(200),xincr(210)
common/blk2/p,q/blk3/c,d,xincr
z=r**2-125.**2
xint=abs(sqrt(abs(z)))
d(0)=-1.
i=0
do 125 i=1,126
d(i)=d(i-1)+1.
zz=((r**2)-(d(i)**2))
zz=abs(zz)
c(i)=abs(sqrt(zz))
xincr(i)=c(i)-xint
continue
do 126 lo=1,(kk-1)
g(lo)=p(lo)
continue
do 127 l=1,(kk-1)
w=abs(q(l))
do 127 n=1,126
if(abs(w-d(n)).le.0.51) g(l)=g(l)+xincr(n)
if(w.eq.0.0) go to 127
if(abs(g(l)-p(l)).eq.0.0) go to 127
diff=abs(g(l)-p(l)-xincr(n)-xincr(n-1))
if(diff.lt.0.00000001) g(l)=g(l)-xincr(n)
continue
do 128 lo=1,(kk-1)
```

```
p(lo)=g(lo)
continue
return
end

subroutine shiyy(rr,kk,g)
dimension c(210),d(210),p(200),q(200),g(200),xincr(210)
common/blk3/c,d,xincr/blk2/p,q
z=rr**2-200.**2
yint=abs(sqrt(abs(z)))
c(0)=-1.
i=0
do 175 i=1,201
c(i)=c(i-1)+1.
zz=((rr**2)-(c(i)**2))
zz=abs(zz)
d(i)=abs(sqrt(zz))
xincr(i)=d(i)-yint
continue
do 176 lo=1,(kk-1)
g(lo)=q(lo)
continue
do 177 l=1,(kk-1)
ww=abs(p(l))
do 177 n=1,201
if(abs(ww-c(n)).le.0.51) g(l)=g(l)+xincr(n)
if(ww.eq.0.0) go to 177
if(abs(g(l)-q(l)).eq.0.0) go to 177
diff=abs(g(l)-q(l)-xincr(n)-xincr(n-1))
if(diff.lt.0.00000001) g(l)=g(l)-xincr(n)
continue
do 178 lo=1,(kk-1)
q(lo)=g(lo)
continue
return
end

subroutine sket(kk)
dimension p(200),q(200)
common/blk2/p,q
jz=1
call move(p(jz),q(jz))
jz=jz+1
call line(p(jz),q(jz))
if(jz.eq.(kk-1)) go to 61
go to 60
return
end

subroutine dist3(x,y,no)
dimension x(50),y(50),a(50),b(50)
common/blk1/a,b
do 302 i=1,no
x(i)=1.5*a(i)
y(i)=b(i)
continue
return
end

subroutine dist5(x,y,no)
dimension x(50),y(50),a(50),b(50)
common/blk1/a,b
do 303 i=1,no
x(i)=a(i)*0.8
y(i)=b(i)*1.6
continue
return
end

subroutine dist4(x,y,no)
dimension x(50),y(50),a(50),b(50)
common/blk1/a,b
```

```
      dimension x(50),y(50),a(50),b(50)
      common/blk1/a,b
      do 304 i=1,no
      x(i)=1.6*b(i)
      y(i)=0.8*a(i)
      continue
      return
      end

      subroutine dist1(x,y,no)
      dimension x(50),y(50),a(50),b(50)
      common/blk1/a,b
      do 310 i=1,no
      x(i)=a(i)+b(i)*sin(3.14/6)/cos(3.14/6)
      y(i)=b(i)
      continue
      return
      end

      subroutine dist7(x,y,no)
      dimension x(50),y(50),a(50),b(50)
      common/blk1/a,b
      do 311 i=1,no
      x(i)=1.2*a(i)+b(i)*(sin(0.5235)/cos(0.5235))
      y(i)=0.8*b(i)+a(i)*(sin(0.5235)/cos(0.5235))
      continue
      return
      end

      subroutine dist6(x,y,no)
      dimension x(50),y(50),a(50),b(50)
      common/blk1/a,b
      do 312 i=1,no
      x(i)=a(i)
      y(i)=b(i)+a(i)*sin(0.5235)/cos(0.5235)
      continue
      return
      end

      subroutine dist2(x,y,no)
      dimension x(50),y(50),a(50),b(50)
      common/blk1/a,b
      do 315 i=1,no
      x(i)=1.5*a(i)
      y(i)=1.3*b(i)
      continue
      return
      end
```