* Read two end points say A(x1,y1) & B(x2,y2).
* Read lower left corner & upper right corner of clipping window: wxmin, wxmax, wymin, wymax.
* Assign 4 bit region code to A & B.

code: B4 B3 B2 B1

0110

0100

0101

0010

0000

0001

1010

1000

1001

* If point is left of window i.e. x<wxmin, then B1=1 else B1=0
* If point is right of window i.e. x>wxmax, then B2=1 else B2=0
* If point is below the window i.e. y<wymin, then B3=1 else B3=0
* If point is above the window i.e. y>wymax, then B4=1 else B4=0
  + - To check whether line falls in clipping categories:

If both end point code value is 0000 then line seg is inside and display it.

* + - If logical AND operation of both end point code value is not 0000 then line seg is completely outside. Discard it & stop.
    - If logical AND operation of both end point code value is 0000 then line has an intersecting edge & needs clipping.
* Determine intersecting boundary.
  + - * If B1=1 then intersect with line x=wxmin
      * If B2=1 then intersect with line x=wxmax
      * If B3 then intersect with line y=wymin
      * If B4 then intersect with line y=wymax
* Determine co-ordinate of intersecting Boundary, say intersect point I(x,y)
* X=xwmin or xwmax

For y, (x-x1)/(x2-x1) = (y-y1)/y2-y1)

y-y1= [(x-x1)/x2-x1)]\*(y2-y1)

Y=y1+m(x-x1)

* Y=wymin or wymax

For x, (x-x1)/x2-x1)=(y-y1)/(y2-y1)

X-x1=[(y-y1)/(y2-y1)]\*(x2-x1)

X=x1+(1/m)(y-y1)

* Replace end point with intersecting point & form new line.
* Display new line.
* Stop