

Cohen - Sutherland Line Clipping Algorithm

```
-Accept the window co-ordinates winmin.x, winmin.y, winmax.x, winmax.y
-Accept the x and y co-ord of the 2 end points p1.x, p1.y, p2.x, p2.y
-done <- false
-draw <- false
-while not done
    - code1 <- getcode(p1)
    - code2 <- getcode(p2)
    - if code1=0000 and code2=0000
        - done <- true
        - draw <- true
    - else
        - if code1 AND code2 <> 0000
            - reject line
            - done <- true
            - draw <- false
        - else
            - Let p1 be the point which is not inside (swap with p2 if required)
            - find slope m
            - if code1(left boundary)=1
                - p1.x <- winmin.x
                - find p1.y using eqn of line p1.y <- p1.y + (winmin.x - p1.x)*m
            - else
```

```
-if code1(right boundary)=1
    - p1.x <- winMax.x
    - p1.y <- p1.y + (winmax.x - p1.x)*m
- else
    - if code1(bottom boundary)=1
        - p1.y <- winmin.y
        - p1.x <- p1.x + (winmin.y - p1.y)/m
    - else
        - if code1(top boundary)=1
            - p1.y <- winmax.y
            - p1.x <- p1.x + (winmax.y - p1.y)/m
- if draw <- true
    - plot line from (p1.x, p1.y) to (p2.x, p2.y)
```