Exercise – Animated point vs static line segment

- We have a point **Bs(-2,2)** that is travelling in one frame time to another point **Be(3,-1)**.
- A bouncy wall is represented by a line segment and located at **L[(0,-3),(0,3)]**
- Find the final point position **Be'** after collision (**if** any?) and reflection of the point with the wall.

Solution – Following the notes printed and given in class

Step1 – Collision detection - Method 2

- Compute N.Bs, N.P0 and N.Be:
  - N.Bs = (6,0).(-2,2) = -12
  - N.P0 = (6,0).(0,-3) = 0
  - N.Be = (6,0).(3,-1) = 18

    N.Bs < N.P0 and N.Be > N.P0

a- Test passed – no rejection

b- Test Passed – no rejection

c- Compute **N.V**

  a. If (**N.V** == 0) then no collision

     N.V is N(BsBe) = N.Be – N.Bs = 18 – (-12) = 30 ≠ 0

     N.**V** is ≠ 0 – no rejection

d- Compute **Ti**, the time of intersection where **Ti** = (**N**.**P0** - **N**.**Bs**) / (**N**.**V**)

   Ti = (0 – (-12)) / (30) = 2/5

e- If (**Ti** < 0) or (**Ti** > 1) then reject

   Ti = (0 <= **2/5** <= 1)

   Bi = **Bs** + **Ti*****V** = (-2,2) + 2/5*(5,-3) = (-2,2) + (2, -6/5) = (0, 4/5)

f- Test to check if **Bi** is within **P0** and **P1** area. We test **if((Bi – P0).(Bi – P1) < 0)** then return collision at point **Bi**

   (Bi – P0).(Bi – P1) = [(0,4/5) - (0,-3)].[(0,4/5) - (0,3)] = (0,19/5).(0,-11/5) = -209/25 **< 0**

If we followed **Method 3** steps, the first two steps would be to compute the **outward** normal of **BsBe** and do the **rejection test** as follow:

a- **M** = Outward normal of **V** = (**V.y**, -**V.x**) = (-3,-5)

b- **(BsP0.M)*(BsP1.M)** = [(0-(-2) , (-3)-2).(-3,-5)]*[((0-(-2) , 3-2).(-3,-5)]
= [(2,-5).(-3,-5)]*[(2,1).(-3,-5)]
= (-6 + 25)*(-6 + (-5)) = 19*(-11) = -209 **< 0**

This means we can proceed and compute **Ti** and **Bi** as in **Method 2**

Step2 – Reflection

a- Compute **Be'**

Be' = Bi + I - 2(I.n)*n

Where **I** is the penetration vector and **n** is **N** normalized

I = **Be** - **Bi** = (3,1) - (0,4/5) = (3, -9/5)

n = **N**/Length(**N**) = (6,0)/6 = (1,0)

⇨ **Be'** = (0,4/5) + (3,-9/5) - 2*[(3,-9/5).(1,0)]*(1,0)

⇨ **Be'** = (0,4/5) + (3,-9/5) - 2*(3)*(1,0)

⇨ **Be'** = (-3, -1)