DPBS(PG) College, Anoopshahr

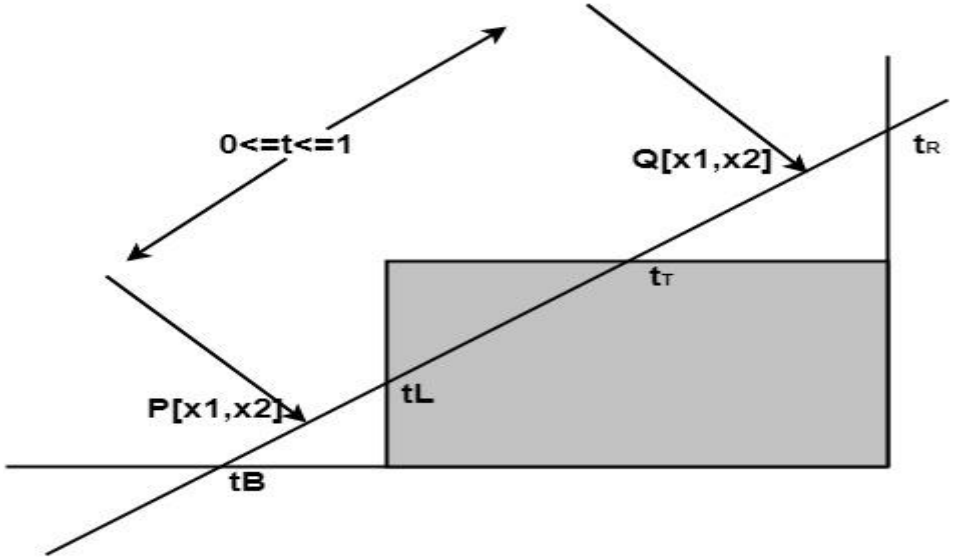
BCA IV Semester

Subject: Computer Graphics

Paper Code: 401

Liang-Barsky Line Clipping Algorithm:

Liang and Barsky have established an algorithm that uses floating-point arithmetic but finds the appropriate endpoints with at most four computations. This algorithm uses the parametric equations for a line and solves four inequalities to find the range of the parameter for which the line is in the viewport.



Let P(x1, y1), Q(x2, y2) is the line which we want to study. The parametric equation of the line segment from gives x-values and y-values for every point in terms of a parameter that ranges from 0 to 1. The equations are

$x=x_1+(x_2-x_1)*t=x_1+dx*t$ and $y=y_1+(y_2-y_1)*t=y_1+dy*t$

We can see that when t = 0, the point computed is P(x1, y1); and when t = 1, the point computed is Q(x2, y2).

1. Set $t_{min}=0$ and $t_{max}=1$

2. Calculate the values t_L, t_R, t_T and t_B(tvalues).

   If tmin or tmax? ignore it and go to the next edge
   Otherwise classify the tvalue as entering or exiting value (using inner product to classify)
   If t is entering value set $t_{min}=t$ if t is exiting value set $t_{max}=t$.

3. If $t_{min} < t_{max?}$ then draw a line from $(x1 + dx*t_{min}, y1 + dy*t_{min})$ to $(x1 + dx*t_{max?}, y1 + dy*t_{max?})$

4. If the line crosses over the window, you will see $(x1 + dx*t_{min}, y1 + dy*t_{min})$ and $(x1 + dx*t_{max?}, y1 + dy*t_{max?})$ are intersection between line and edge.