```
clear
clc


Q_pos1 = 8e-9;          % Point charge 1
Q_pos2 = 8e-9;          % Point charge 2
lambda_line = 4e-9;     % Linear charge density
eps0 = 8.8419e-12;      % Permittivity of free space

obsPoint  = [0 0 0];    % Observation point
pos_Q1    = [0 1 1];    % Location of Q1
pos_Q2    = [0 -1 1];   % Location of Q2
lineCtr   = [3.5 3.5 0];
leftEnd   = [7 0 0];        % One end of line
rightEnd  = [0 7 0];        % Other end of line
nSegments = 1e5;            % Number of segments for integration

vec_Q1 = obsPoint - pos_Q1;
vec_Q2 = obsPoint - pos_Q2;

%distances
dist_Q1 = norm(vec_Q1);
dist_Q2 = norm(vec_Q2);

E_Q1 = (Q_pos1 / (4*pi*eps0*dist_Q1^3)) * vec_Q1;
E_Q2 = (Q_pos2 / (4*pi*eps0*dist_Q2^3)) * vec_Q2;

%%line integration
% Distance from obsPoint to line center (not directly used here, but kept)
dummy_dist = norm(obsPoint - lineCtr);

% Full line's length 7^2 + 7^2
lineLen  = sqrt(7^2 + 7^2);
lineDir  = (rightEnd - leftEnd) / lineLen;          % Unit direction
segVec   = (lineLen / nSegments) * lineDir;         % Vector for each sub-segment
segSize  = norm(segVec);                            % Scalar length of each segment

E_line = [0 0 0];

% Compute center of first segment (shifting half the lineLen * direction, etc.)
segCenter = lineCtr - ((nSegments/2) * segVec - segVec/2);

%%integrate
for idx = 1:nSegments
    R_vec   = obsPoint - segCenter;     % Vector from sub-segment center to obsPoint
    R_mag   = norm(R_vec);
    dE_line = segSize * lambda_line / (4 * pi * eps0 * R_mag^3) * R_vec;
    E_line  = E_line + dE_line;         % Accumulate field
    segCenter = segCenter + segVec;     % Move to the next segment's center
end

E_total = E_Q1 + E_Q2 + E_line
```