Problem 1

Problem 2

Problem 3

Problem 4

1. Merge(S1, S2): L(S1 ∪ S2)  
   create a new array to store the output A  
   i ← 1, j ← 1  
   for k ← 1 to (n1+n2)  
     
    if j > n2 then  
    terminate program   
    if i > n1 then  
    A[k]←S2[j]  
    j←j+1  
      
    if S1[i].y < S2[j].y then  
    i←i+1  
    else if S1[i].y > S2[j].y then  
    A[k]←S1[j]  
    i←i+1  
    else   
    A[k]←S1[i]  
    k←k+1  
    A[k]←S2[j]  
    i←i+1  
    j←j+1
2. Correctness:   
   S1 and S2 are sorted by x-coordinate  
   By property 1(a), the points are sorted by x in ascending order and by y in descending order. Because if for i < j, then and contradict the property.  
     
   if then ignore ,  
   else if then store ,  
   else store both.  
     
   :  
   Go through from k = 1 to k = (n1 + n2)
3. FINDL(S):  
   n ← size of S  
   mid ← n / 2  
   if n = 1 then return S  
   L ← FINDL(S[0 to mid])  
   R ← FINDL(S[mid + 1 to n])  
   return Merge(L, R)
4. T(n) = 2T(n/2) + O(n) = O(nlogn)  
   Correctness:  
   Recursively find L(first half of S) and L(second half of S) and merge them into one.