

Estructuras de Datos y Algoritmos II

Tarea 06

Rubén Pérez Palacios
Profesor: Dr. Carlos Segura González

15 Marzo 2020

La solución del problema es con la estrategia Divide and Conquere. Dado un conjunto P de puntos lo divide en dos subconjuntos PL y PR tal que para todos $p \in PL$, $q \in PR$ se cumple que $p \leq q$ y $|PL| - |PR|$ es a lo más uno; encuentra el número de puntos dominados de conjunto por los puntos del otro, y recursivamente resuelve lo mismo para PL y PR .

Dado los conjuntos PL y PR supongamos ordenados por su coordenada y , podemos calcular los puntos dominados procesando los primeros elementos de ambos conjuntos $p \in PL$ y $q \in PR$, en caso de que p tenga una coordenada y menor que la de q entonces p dominara a todos los puntos de PR en ese momento - debido a como fueron creados PL y PR y como fueron ordenados - y descartamos el punto p , en caso contrario habría que checar que las coordenadas de x fuesen iguales entre p y q de ser así entonces checamos que la coordenada y de p sea mayor a la de q de ser así q dominaría todos los puntos del conjunto PL , y descartamos a q .

El algoritmo anterior la cantidad de divisiones son $O(\log(N))$ y para cada una solo se realiza un recorrido a cada arreglo PL y PR que tiene complejidad $O(N)$ esto asumiendo que ambos son ordenados. Podríamos ordenar ambos arreglos en cada división lo cual tiene una complejidad de $O(N \log(N))$, y entonces nuestra complejidad del algoritmo sería $O(N \log(N)^2)$, pero utilizando la técnica del merge-sort podemos mantener ambos conjuntos PL y PR ordenados en cada división con una complejidad de $O(N)$, por lo tanto nuestro algoritmo tiene complejidad

$$O(N \log(N))$$