MAC 331 - LISTA 1

DANIELA GONZALEZ FAVERO - 10277443

(2) A bobagem do esquema proposto pelo professor está no fato de que, dada uma determinada entrada, o algoritmo perde a função de divisão e conquista.
Suponha que todos os pontos estejam alinhados numa reta na vertical: todos os pontos sempre estarão na reta separadora e serão mandados para o conjunto E, além do conjunto D permanecer vazio. Desta forma, o algoritmo não é capaz de dividir o problema e, consequentemente, de resolvê-lo.

(4) Primeiro alteramos a subrotina candidatos:

```
Candidatos (X, a, p, r, d)
    q ← ⌊(p+r)/2⌋
    t ← 0
    u ← 0      ▷ Contador do outro vetor (g)
    para k ← p até r faça
        se |X[a[k]] - X[q]| < d
            então se X[a[k]] ≤ X[q] ▷ está à esquerda
                então t ← t+1
                      f[t] ← a[k]
                senão u ← u+1  ▷ está à direita
                      g[u] ← a[k]
    devolva (f, t, g, u)
```

Note que a complexidade dessa subrotina se mantém a mesma pois não alteramos o laço - só adicionamos outra condição (de tempo constante) dentro do laço.

Agora vamos usar a informação dos vetores f e g para alterar a rotina Combine:

```
Combine (X, Y, a, p, r, dE, dD)
    d ← min {dE, dD}
    (f, t, g, u) ← Candidatos (X, a, p, r, d)
    para i ← 1 até t - 1 faça
        para j ← 1 até min {i + 4, u} faça
            d' ← Dist (X[f[i]], Y[f[i]], X[g[j]], Y[g[j]])
            se d' < d
                então d ← d'
    devolva d
```

Note que em cada um dos dois quadrados de lado d, há no máximo 4 pontos porque $d \leq d_E$ e $d \leq d_D$; por isso o segundo laço percorre até $\min\{i+4, u\}$.

A complexidade do algoritmo se manteve, ele ainda é linear, mas consome $4n$ em vez de $7n$. O gargalo continua sendo a ordenação, que consome $n \lg n$.