**Descrição do Problema e da Solução**

Para ambos os problemas para descobrir o tamanho da maior subsequência possível, a ideia é ter um vetor ­(lensList) em que na mesma posição de um certo elemento teria o tamanho máximo de uma subsequência crescente que acaba nesse elemento. Para o problema 1 teremos uma lista adicional para o numero de subsequências crescentes que existem com o tamanho indicado na posição com o mesmo índice na lensList.

**Análise Teórica**

* Leitura dos dados de entrada: simples leitura do input, com ciclo(s) a depender linearmente O(n)
* Processamento da instância para fazer alguma coisa. Logo, O(1)
* Aplicação do algoritmo 1 para encontrar a numero e o tamanho das maiores subsequências crescentes numa lista com n elementos. Logo, O(n^2)
* Aplicação do algoritmo 2 para encontrar tamanho da maior subsequência comum crescente entre uma lista com n elementos e outra com m elementos. Logo, O(n\*m)
* Calculo do numero de listas com o maior tamanho possível. O(n^2)
* Apresentação dos dados. O(1)

Complexidade da solução do 1º algoritmo: O(n^2)

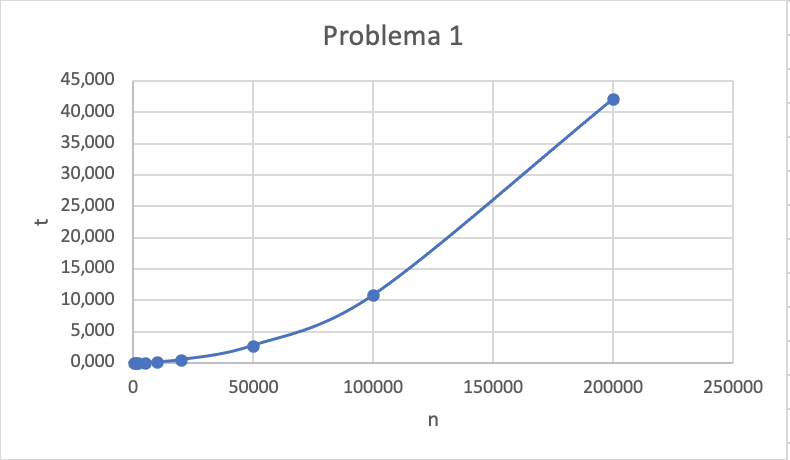
Complexidade da solução do 2º algoritmo: O(n\*m)

**Avaliação Experimental dos Resultados**

Problema 1:

* Input1 n=100 t=0,006s
* Input2 n= 500 t=0,007s
* Input3 n=1 000 t=0,009s
* Input4 n=2 000 t=0,017s
* Input5 n=5 000 t=0,060s
* Input6 n=10 000 t=0,149s
* Input7 n=20 000 t=0,480s
* Input8 n=50 000 t=2,765s
* Input9 n=100 000 t=10,813s
* Input10 n=200 000 t=42,155s

Eixo dos YY’s refere-se ao tempo do algoritmo e o eixo dos XX’s refere-se ao numero de elementos da lista



Problema 2:

* Input1 n=100 m=100 t=0,006s
* Input2 n=500 m=500 t=0,008s
* Input3 n=1 000 m=1 000 t=0,011s
* Input4 n=2 000 m=2 000 t=0,039s
* Input5 n=5 000 m=5 000 t=0,101s
* Input6 n=10 000 m=10 000 t=0,262s
* Input7 n=20 000 m=20 000 t=1,442s
* Input8 n=50 000 m=50 000 t=5,606s
* Input9 n=100 000 m=100 000 t=22,166s
* Input10 n=200 000 m=200 000 t=87,540s

Eixo dos YY’s refere-se ao tempo do algoritmo e o eixo dos XX’s refere-se ao numero de elementos de uma das listas

