

Sim.

Não só é legal, como vai nos permitir escapar do dilema.

Quer dizer, utilizando a intercalação esperta, a situação que nós temos com $\sqrt{n}$ blocos é

$$\text{ordenação: } \sqrt{n} \cdot (\sqrt{n})^2 = n \cdot \sqrt{n}$$

$$\text{intercação: } \log(\sqrt{n}) \cdot n = \frac{1}{2} n \cdot \log n$$

Quer dizer, os tempos de ordenação e intercalação não são mais iguais.

E isso significa que a gente pode continuar diminuindo o tamanho dos blocos.

Mas, até quando a gente pode diminuir?

Bom, agora vai acontecer uma coisa engraçada ...

Quer dizer, imagine que a lista é dividida em blocos de tamanho $\sqrt[4]{n} = n^{1/4}$

De modo que o número total de blocos é $n^{3/4}$ — (*porque?*)

Daí, a situação fica assim

$$\text{ordenação: } n^{3/4} \cdot (n^{1/4})^2 = n \cdot \sqrt[4]{n}$$

$$\text{intercação: } \log(n^{3/4}) \cdot n = \frac{3}{4} n \cdot \log n$$

Mas,

$$\frac{3}{4} \log n < \sqrt[4]{n}$$

— (*porque?*)

Quer dizer, o tempo de ordenação continua maior do que o tempo de intercalação.

Logo, a gente pode continuar diminuindo o tamanho dos blocos.

Imagine agora que nós dividimos a lista em blocos de tamanho $\sqrt[8]{n} = n^{1/8}$

— (*com um total de $n^{7/8}$ blocos*)

Daí, a situação fica assim

$$\text{ordenação: } n^{7/8} \cdot (n^{1/8})^2 = n \cdot \sqrt[8]{n}$$

$$\text{intercação: } \log(n^{7/8}) \cdot n = \frac{7}{8} n \cdot \log n$$

Mas,

$$\frac{7}{8} \log n < \sqrt[8]{n}$$

E o tempo de ordenação continua maior do que o tempo de intercalação.

Você já percebeu o que está acontecendo?

Sim! nós podemos diminuir o tamanho dos blocos para sempre.

E fazendo isso, a gente chega em blocos de tamanho 1.

Só que um bloco de tamanho 1 não precisa ser ordenado.

Quer dizer, não vai mais ter uma etapa de ordenação.

Só o que vai ter é intercalação.

Em outras palavras, nós acabamos de obter o algoritmo de *ordenação por intercalação*.

Esse algoritmo divide a lista em blocos de tamanho 1.

E o seu tempo de execução é $O(n \log n)$.