

Análise de algoritmos

SCC-214 Projeto de Algoritmos

Thiago A. S. Pardo



Análise de algoritmos

- Precauções

- Alguns dizem que a **expressão correta** é “ $f(n)$ é $O(g(n))$ ”
 - Seria considerado **redundante e inadequado** dizer “ $f(n) \leq O(g(n))$ ” ou (ainda pior) “ $f(n) = O(g(n))$ ”
 - **Não é incorreto** (embora não seja usual) dizer “ $f(n) \in O(g(n))$ ”, já que o operador *Big-oh* representa todo um conjunto de funções

2

Precauções



- A **análise assintótica é uma ferramenta fundamental ao projeto**, análise ou escolha de um algoritmo específico para uma dada aplicação
- No entanto, deve-se ter sempre em mente que essa análise “esconde” fatores assintoticamente irrelevantes, mas que em alguns casos podem ser relevantes na prática, particularmente se o problema de interesse se limitar a entradas (relativamente) pequenas
 - Por exemplo, um algoritmo com tempo de execução da ordem de $10^{100}n$ é $O(n)$, assintoticamente melhor do que outro com tempo $10 n \log n$, o que nos faria, em princípio, preferir o primeiro
 - No entanto, 10^{100} é o número estimado por alguns astrônomos como um limite superior para a quantidade de átomos existente no universo observável!

3

Exercício



- Problema da maior soma de subsequência em um vetor

-2	11	-4	13	-5	-2
$\underline{\hspace{1.5cm}}$ 20					

- Implemente um programa para resolver o problema e analise-o

4

Exercício



- Maior soma de subsequência em um vetor
 - Algoritmo de $O(n^3)$

5

Exercício



- Maior soma de subsequência em um vetor
 - Algoritmo de $O(n^2)$

6

Exercício



- Maior soma de subsequência em um vetor
 - Algoritmo de $O(n \log n)$

7

Exercício



- Maior soma de subsequência em um vetor
 - Algoritmo de $O(n)$

8

Exercício



- Implemente um programa para calcular x^n e analise-o

9

Exercício



- Exponenciação
 - $O(n)$

10

Exercício



- Exponenciação
 - $O(\log n)$

11

Exercício



- Análise do algoritmo de Euclides
 - Um dos algoritmos mais antigos conhecidos (300 anos antes de Cristo)
 - Cômputo do máximo divisor comum entre dois números m e n , com $m \geq n$
 - $\text{mdc}(50, 15) = 5$

```
int mdc(int m, int n) {
    int aux=0;
    while (n>0) {
        aux=m%n;
        m=n;
        n=aux;
    }
    return(m);
}
```

12