FCT/Unesp – Presidente Prudente
Departamento de Matemática e Computação

Notação Assintótica
Parte I

Prof. Danilo Medeiros Eler
danilo.eler@unesp.br

Apresentação adaptada (ver referências)

Departamento de Matemática e Computação
Faculdade de Ciências e Tecnologia
Unesp – Universidade Estadual Paulista
Presidente Prudente/SP, Brasil

Análise de Algoritmos

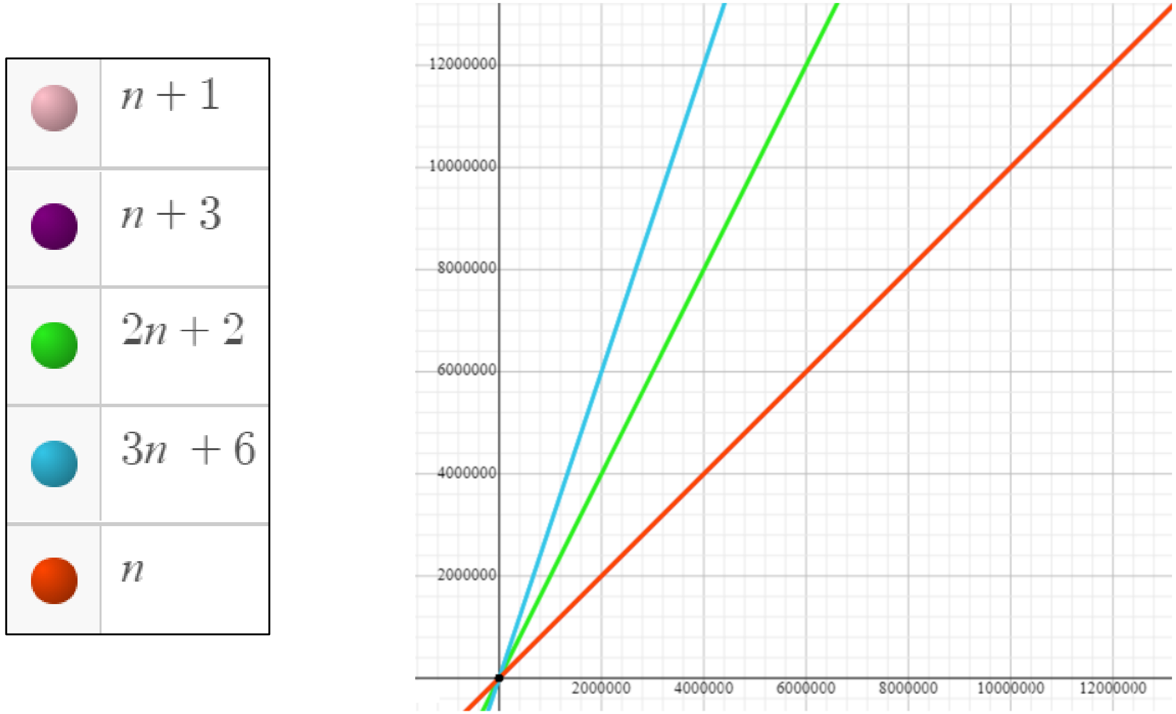
```
Pessoa busca(String nome){
   for (int i = 0; i< pessoas.size(); i++){
         if (pessoas.get(i).getNome().equals(nome))
             return pessoas.get(i);
   }
   return null;
}
```

Melhor Caso: 2   Pior Caso: n + 1

Análise Assintótica: $\Omega(1)$ $O(n)$

Análise Assintótica

- Nessa análise, as funções são classificadas em “ordens” e todas aquelas que são de uma mesma ordem são “equivalentes”



Análise Assintótica

```
int menor = MAIOR-INTEIRO;
for (int i=0; i<n; i++)
  if (vetor[i] < menor)
     menor = vetor[i];
if (menor < 0){
   for (int i=0; i<n; i++)
        menor = menor * (i+1);
}else if (menor > 0){
    for (int i=0; i<n*n; i++)
       printf("%d\n", menor);
  } else {
        printf("%d\n", menor);
     }
```

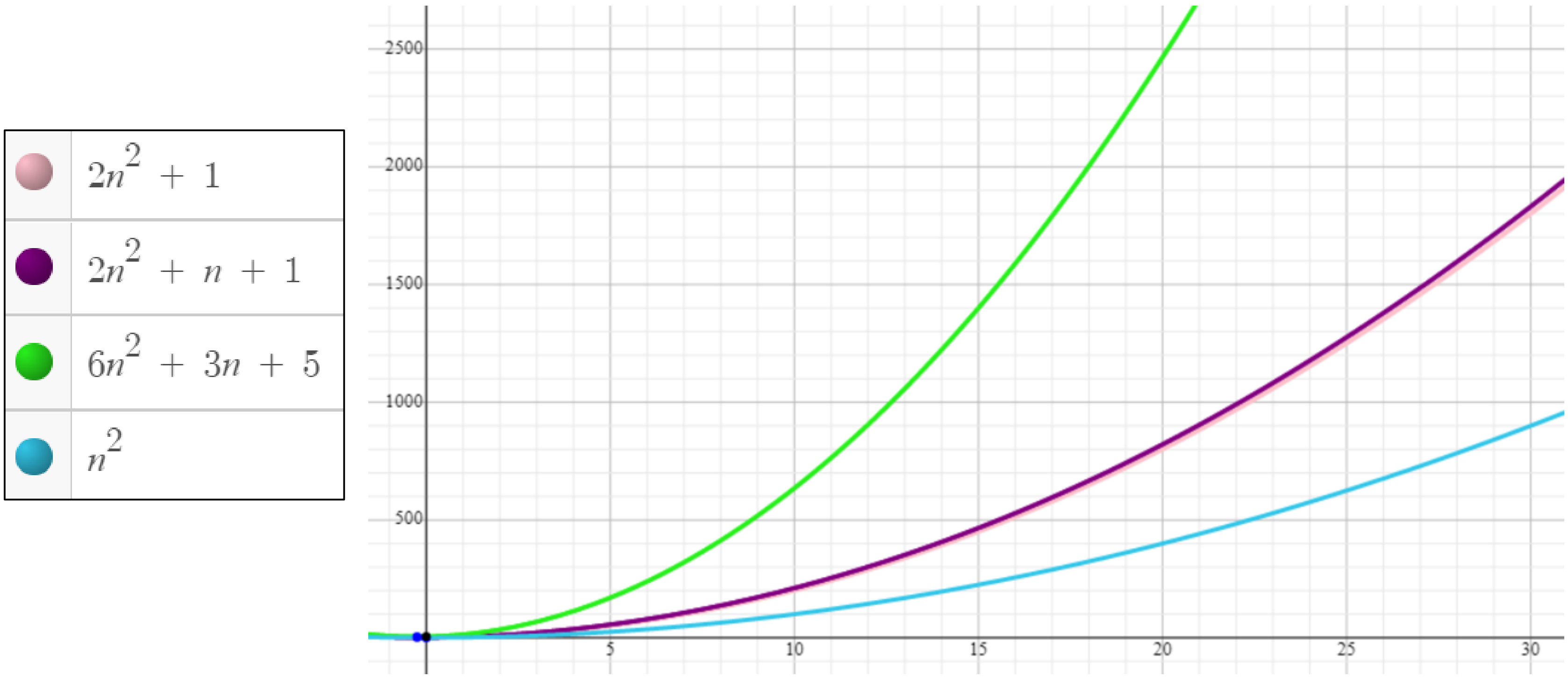
Melhor Caso: $5 + n$

Pior Caso: $3+2n+n^2$

Análise Assintótica: $\Omega(n)$ e $O(n^2)$

Análise Assintótica

- Nessa análise, as funções são classificadas em “ordens” e todas aquelas que são de uma mesma ordem são “equivalentes”



Análise Assintótica

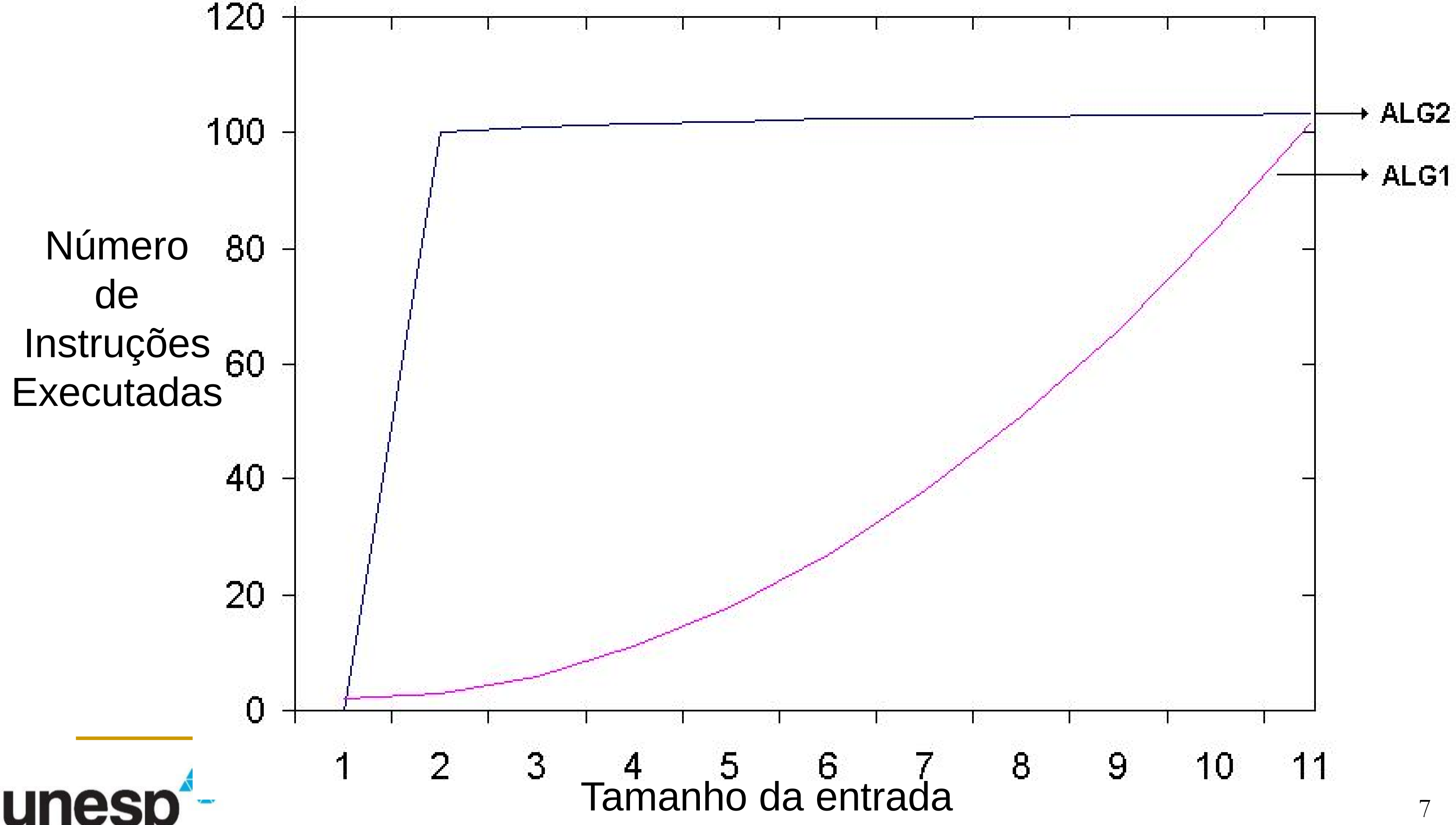
- Em geral, um algoritmo que é assintoticamente mais eficiente será a melhor escolha para todas as entradas, exceto para as muito pequenas
- Exemplo
  - Suponha que você precise escolher um entre dois algoritmos de ordenação, considerando o pior caso
    - ALG1 com custo total de $n^2+2 = O(n^2)$
    - ALG2 com custo total de $n\log(n)+100 = O(n\log n)$

unesp

DEPARTAMENTO DE MATEMÁTICA E COMPUTAÇÃO

Análise Assintótica

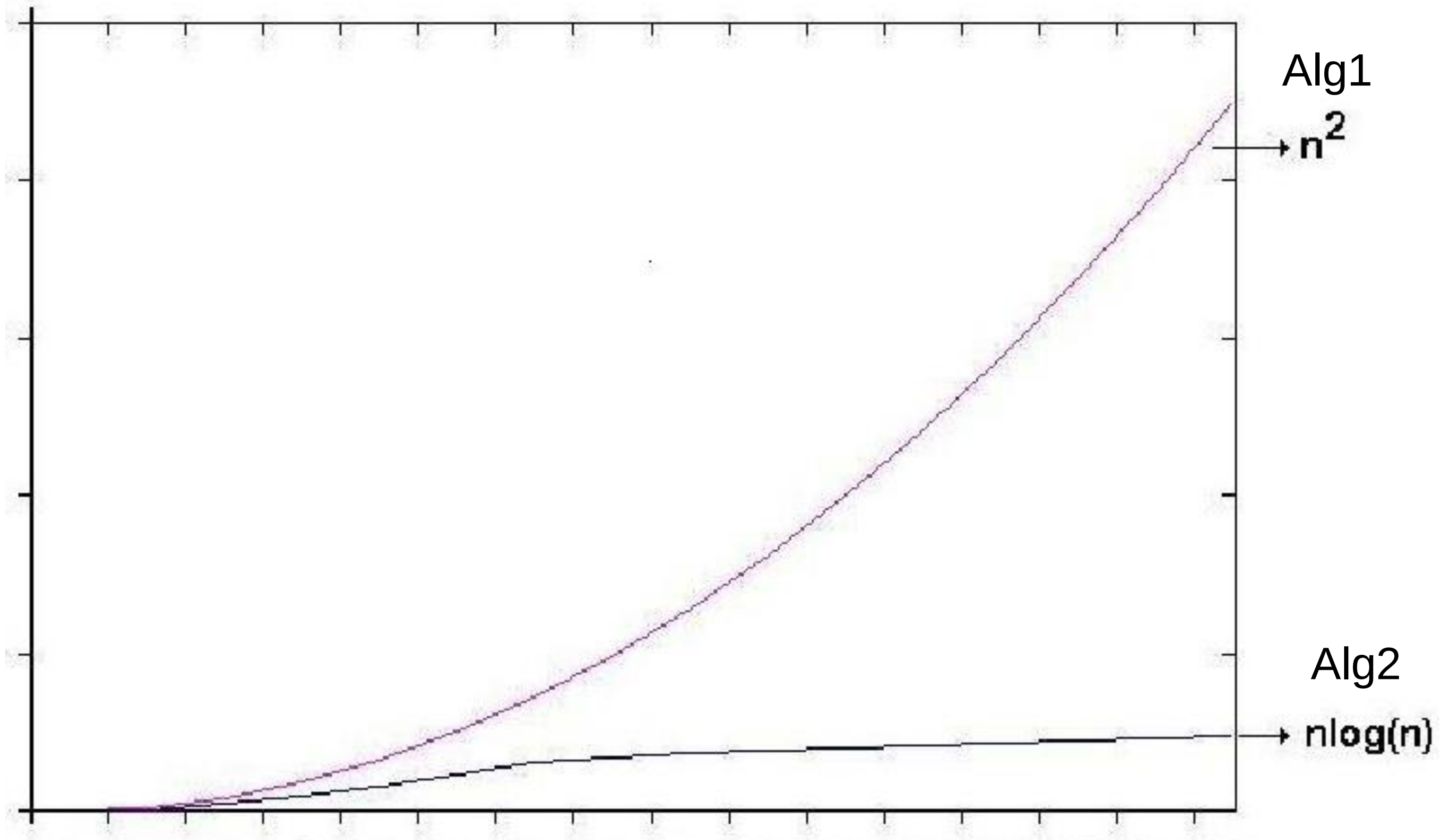
- Entrada pequena (Análise Experimental)



Análise Assintótica

- Análise assintótica

Número
de
Instruções
Executadas



Tamanho da entrada

Notação Assintótica

- A medida de custo ou medida de complexidade relata o crescimento assintótico da operação considerada
- Notação *Big O*
  - Será utilizada para descrever um limite superior para a ordem de complexidade das funções
  - Exemplos
    - $n^2+2 = O(n^2)$
    - $n\log(n)+100 = O(n\log n)$
    - $2n+10 = O(n)$

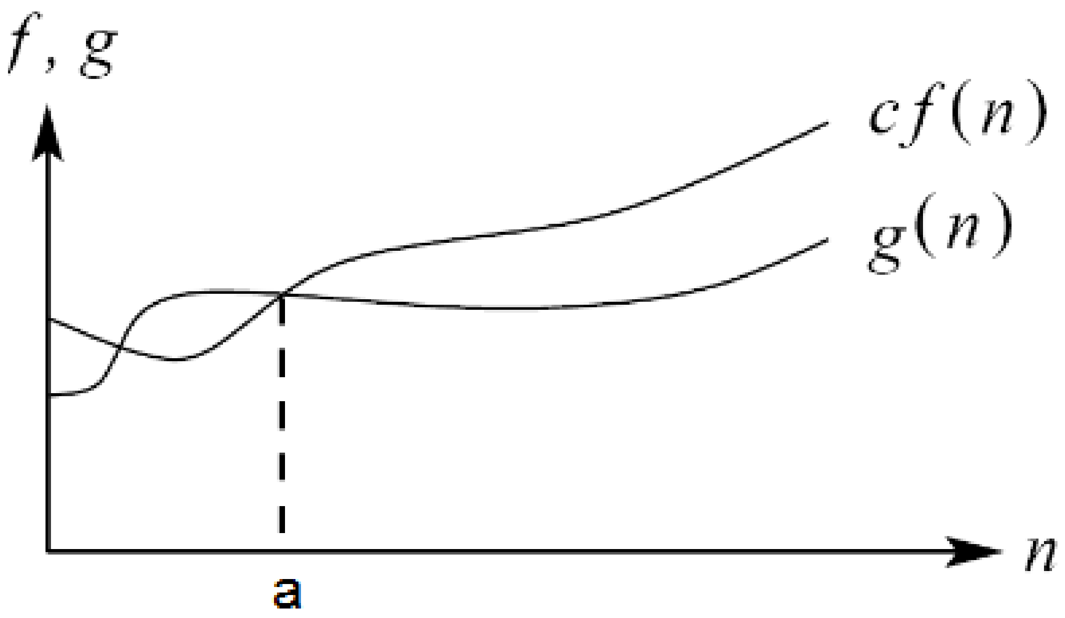
Notação Assintótica

- Notação *Big O*
  - **Definição:** Uma função *f(n)* domina assintoticamente outra função *g(n)* se
    - existem duas constantes positivas ***c*** e ***a*** tais que, para qualquer *n* >= *a*, temos g*(n)* <= *cf(n)*
    - Utilizando a notação, podemos escrever
      - *g(n) = O(f(n))*

Observação: em alguns livros a constante ***a*** será escrita como n_o ou x_o

Notação Assintótica

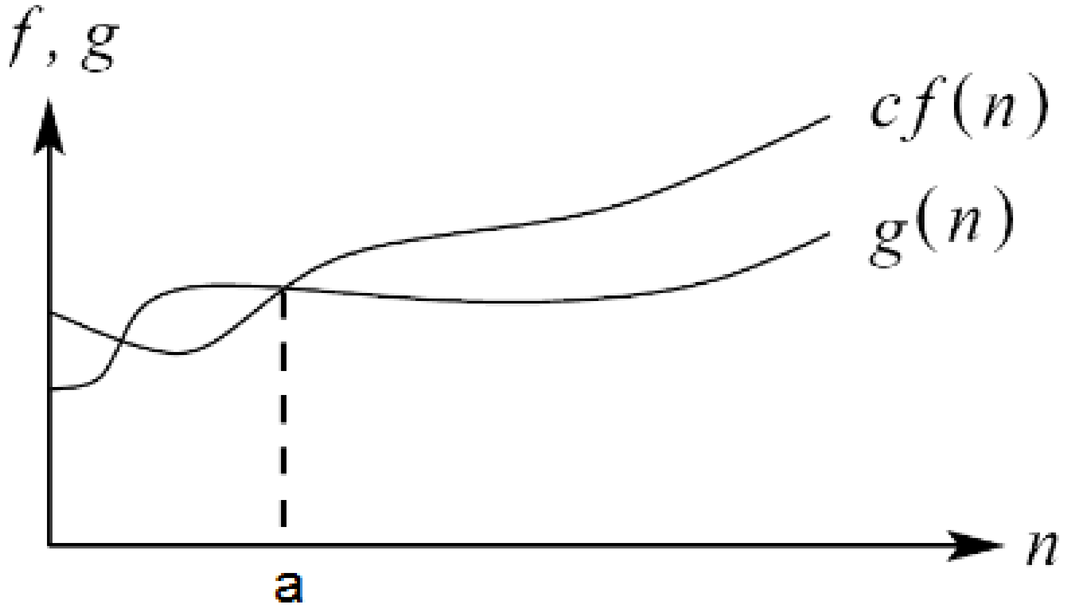
Escrevemos $g(n) = O(f(n))$ ou $g(n) \in O(f(n))$ para expressar que $f(n)$ domina assintoticamente $g(n)$.



Notação Assintótica

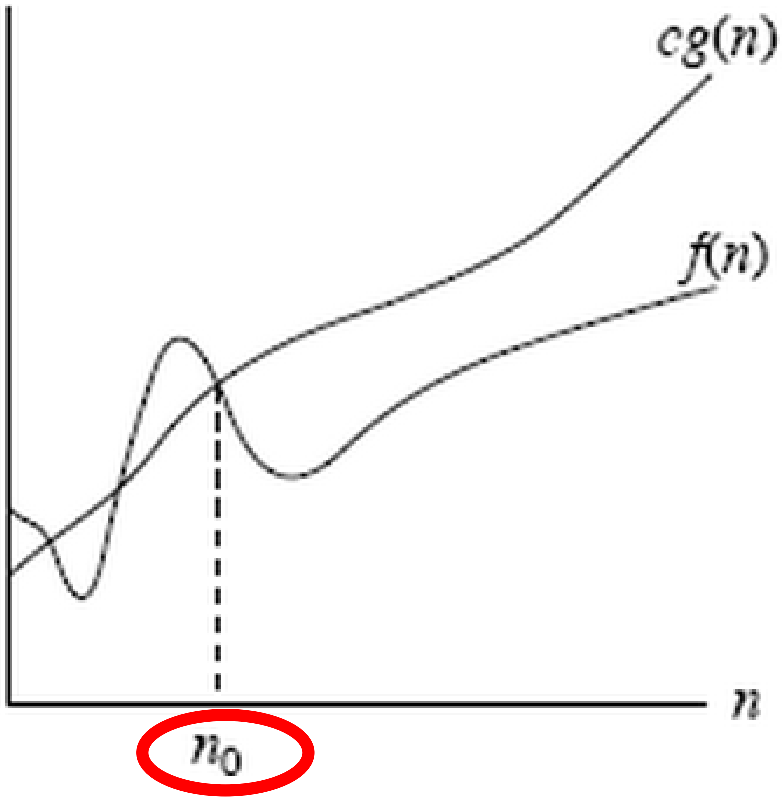
Escrevemos $g(n) = O(f(n))$ ou $g(n) \in O(f(n))$ para expressar que $f(n)$ domina assintoticamente $g(n)$.

Isto é, uma função *g(n)* é *O(f(n))* se após um determinado ponto ***a*** *g(n)* não é maior que *cf(n)*



Notação Assintótica

- Na literatura é comum encontrar n_o ou x_o



Notação Assintótica

- Escrevemos $g(n) = O(f(n))$ ou $g(n) \in O(f(n))$ para expressar que $f(n)$ domina assintoticamente $g(n)$.
- Encontramos leituras nos modos:
  - *g(n)* é da ordem no máximo *f(n)*; *// formal*
  - *g(n)* é *O* de *f(n)*; *// informal*
  - *g(n)* é igual a *O* de *f(n)*; *// informal*
  - *g(n)* pertence a *O* de *f(n)*; *// formal*

Notação Assintótica

- Escrevemos $g(n) = O(f(n))$ ou $g(n) \in O(f(n))$ para expressar que $f(n)$ domina assintoticamente $g(n)$.
- **Exemplo:** quando dizemos que o tempo de execução $T(n)$ de um programa é $O(n^2)$, ou $T(n) = O(n^2)$, significa que
  - existem constantes ***c*** e ***a*** tais que, para valores de $n >= a$, temos:
    - $T(n) <= cn^2$

Notação Assintótica

- Questão 1:
  - n é $O(n^2)$?

Notação Assintótica

- Questão 1:
  - n é $O(n^2)$?

- Sim
- Para c = 1 e a = 1; $n >= a$
  - $n <= cn^2$

DEPARTAMENTO DE MATEMÁTICA E COMPUTAÇÃO

Notação Assintótica

- Questão 2
  - n^2 é $O(n)$?

Notação Assintótica

- Questão 2
  - n^2 é $O(n)$?

- Não
- Suponha: $\exists c, a$

$$\forall n \geq a$$

$$n^2 \leq c \cdot n$$

Análise Assintótica

- Exemplo de funções da ordem O(n^2)
  - $(3/2)n^2$
  - $9999n^2$
  - $n^2/1000$
  - n^2+100n

- Posso realmente afirmar que
  - $n^2+100n = O(n^2)$?

Análise Assintótica

- $n^2+100n = O(n^2)$?
  - Existe uma constante ***c*** e uma constante ***a*** que satisfaz
    - $n^2+100n \le cn^2$, *para valores de* $n \ge a$

Análise Assintótica

- Considerando c = 100
  - $n^2+100n <= cn^2$
  - $n^2+100n <= 100n^2$
  - $(n^2+100n)/n^2 <= 100$
  - $(\cancel{n}^2+100\cancel{n})/\cancel{n}^2 <= 100$
  - $1+100/n <= 100$
  - $100/n <= 99$

  - Fica mais fácil de ver que a afirmação é verdadeira

Análise Assintótica

- *Para n = 1*
  - 100/n <= 99
  - 100/1 <= *99*
  - 100 <= *99* ----- *FALSO*

Análise Assintótica

- *Para n = 1*
  - 100/n <= 99
  - 100/1 <= *99*
  - 100 <= *99* *----- FALSO*

- *Para n = 2*
  - 100/n <= 99
  - 100/2 <= 99
  - 50 <= 99 *----- VERDADEIRO*

Análise Assintótica

- *Para n = 3*
  - 100/n <= 99
  - 100/3 <= *99*
  - 33,33 <= *99 ----- VERDADEIRO*

- *Para n = 4*
  - 100/n <= 99
  - 100/4 <= *99*
  - 25 <= *99 ----- VERDADEIRO*

Análise Assintótica

- $n^2+100n = O(n^2)$?
  - Existe uma constante ***c*** e uma constante ***a*** que satisfaz
    - $n^2+100n <= cn^2$, *para valores de n >= a*
- A afirmação é verdadeira para
  - c = 100
  - a = 2
    - n >= 2

Análise Assintótica

- Considerando c = 10
  - $n^2+100n <= cn^2$
  - $n^2+100n <= 10n^2$
  - $(n^2+100n)/n^2 <= 10$
  - $(\cancel{n}^2+100\cancel{n})/\cancel{n^2} <= 10$
  - $1+100/n <= 10$
  - 100/n <= 9

Análise Assintótica

- *Para n = 1*
  - 100/n <= 9
  - 100/1 <= *9*
  - 100 <= *9 ----- FALSO*

Análise Assintótica

- *Para n = 1*
  - 100/n <= 9
  - 100/1 <= *9*
  - 100 <= *9* ----- *FALSO*
- *Para n = 2*
  - 100/n <= 9
  - 100/2 <= 9
  - 50 <= 9 ----- FALSO

Análise Assintótica

- *Para n = 3*
  - 100/n <= 9
  - 100/3 <= *9*
  - 33,33 <= *9 ----- FALSO*

- *Para n = 4*
  - 100/n <= 9
  - 100/4 <= *9*
  - 25 <= *9 ----- FALSO*

Análise Assintótica

- *Para n = 10*
  - 100/n <= 9
  - 100/10 <= *9*
  - 10 <= *9   ----- FALSO*

- *Para n = 11*
  - 100/n <= 9
  - 100/11 <= *9*
  - 9,09 <= *9   ----- FALSO*

Análise Assintótica

- *Para n = 12*
  - 100/n <= 9
  - 100/12 <= *9*
  - 8,33 <= *9* ----- *VERDADEIRO*

- *Para n = 13*
  - 100/n <= 9
  - 100/13 <= *9*
  - 7,69 <= *9* ----- *VERDADEIRO*

Análise Assintótica

- $n^2+100n = O(n^2)$?
  - Existe uma constante ***c*** e uma constante ***a*** que satisfaz
    - $n^2+100n <= cn^2$, *para valores de* $n >= a$
- A afirmação é verdadeira para
  - c = 10
  - a = 12
    - n >= 12

Notação Assintótica

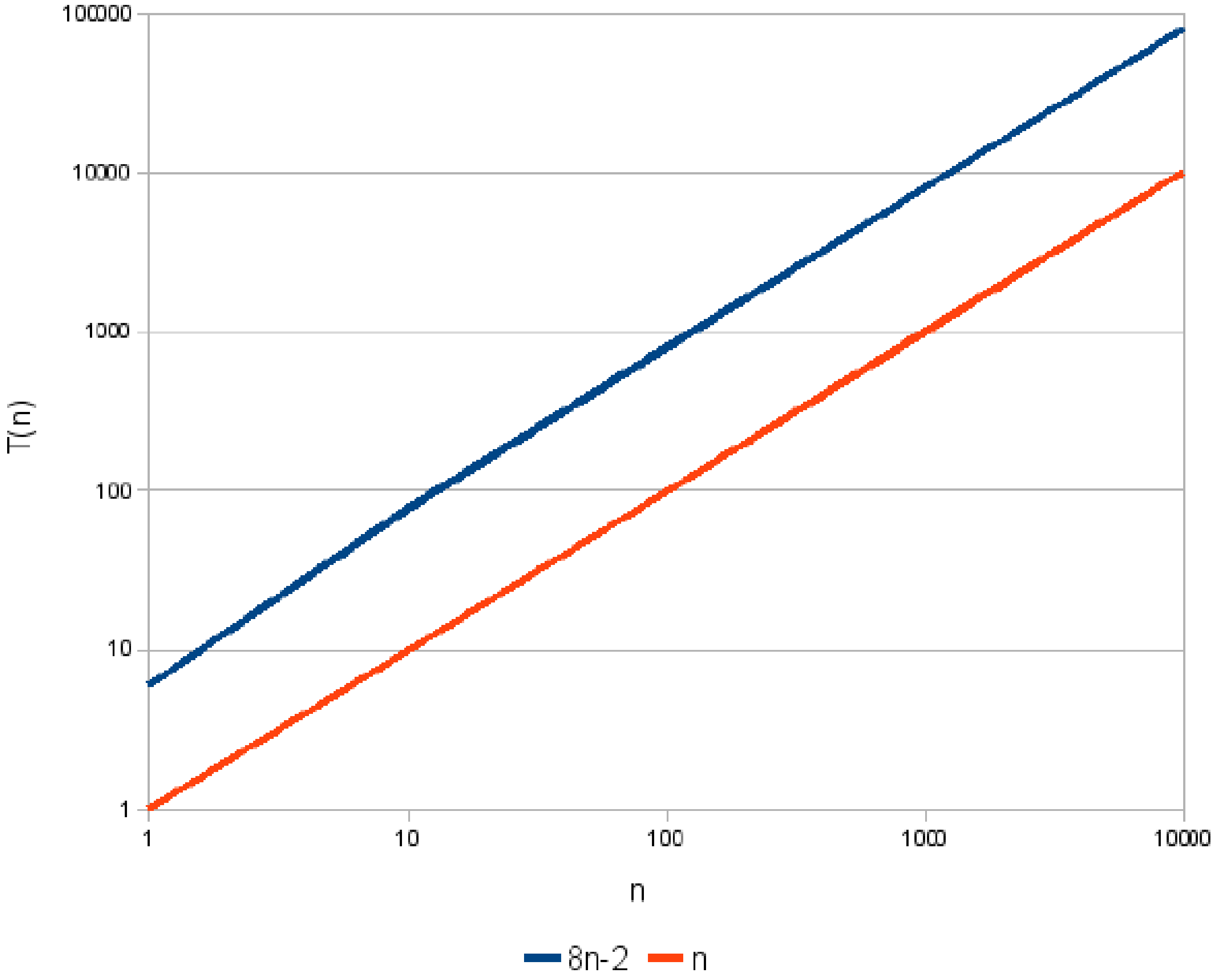
- Estamos interessados nos termos de maior ordem
- Podemos desprezar fatores constantes que o comportamento assintótico da função continua o mesmo

$T(n)$	1	10	100	1.000	10.000
$8n-2$	6	78	798	7.998	79.998
$8n$	8	80	800	8.000	80.000
n	1	10	100	1.000	10.000

unesp

DEPARTAMENTO DE MATEMÁTICA E COMPUTAÇÃO

Notação Assintótica

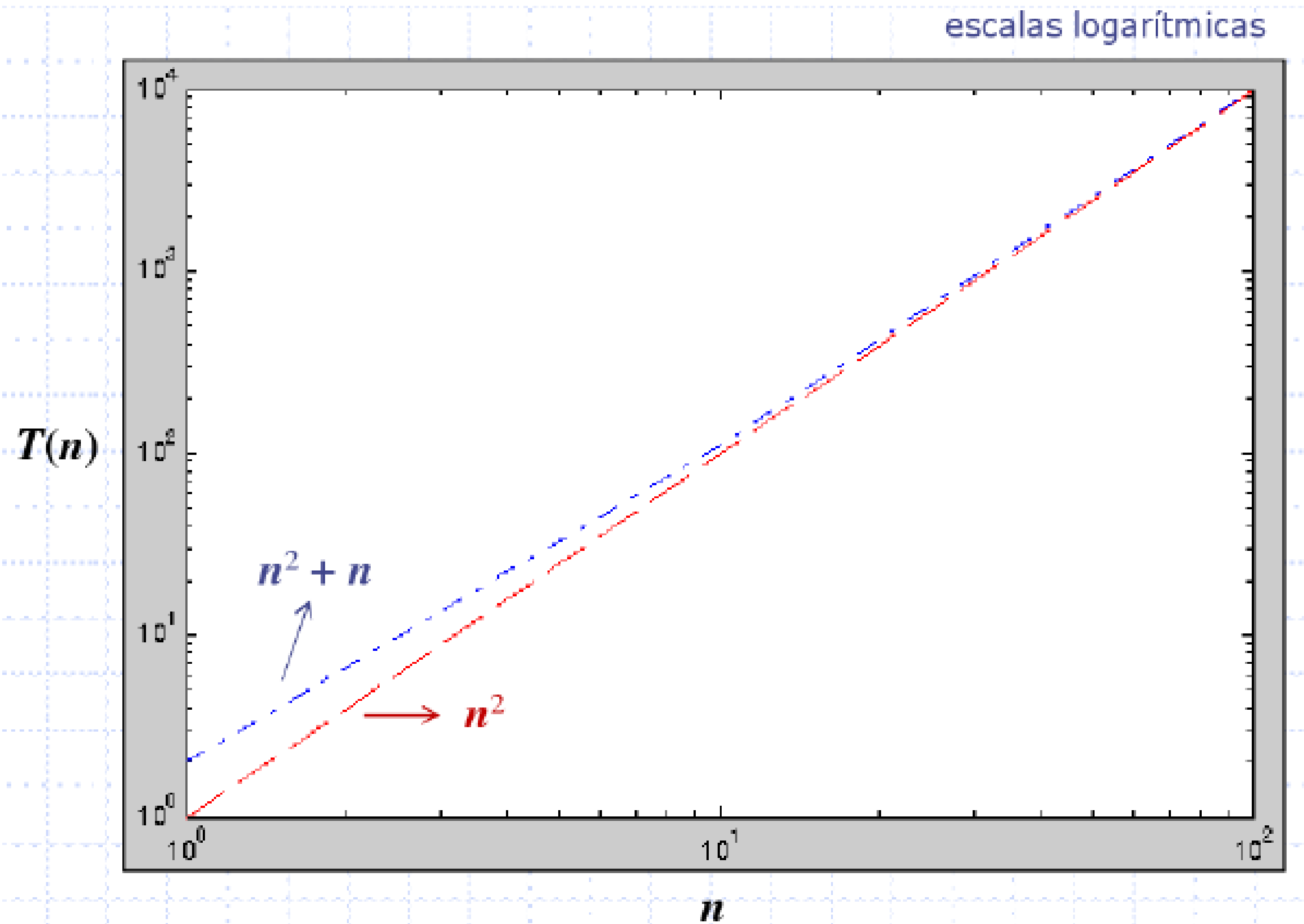


Notação Assintótica

- Também podemos desprezar termos de menor ordem

$T(n)$	1	10	100	1.000
n^2	1	100	10.000	1.000.000
$n^2 + n$	2	110	10.100	1.001.000
Δ	100%	10%	1%	0,1%

Notação Assintótica



Fonte da figura: notas de aula do Prof. Ricardo Campello

Notação Assintótica

- Simplificando
  - Se $f(n)$ é uma função polinomial de grau d, então $f(n)$ é $O(n^d)$, isto é:
    - Descarte termos de menor ordem
    - Descarte termos constantes

Notação Assintótica

- Use a expressão mais simples para representar a classe de complexidade
  - “$3n + 5$ é $O(n)$” em vez de “$3n + 5$ é $O(3n + 5)$”
  - “n^2+100n é $O(n^2)$” *em vez de* “n^2+100n é $O(n^2+100n)$”
  - “n^2+n é $O(n^2)$” *em vez de* “n^2+n é $O(n^2+n)$”

Notação Assintótica

- Operações básicas

$$
\begin{aligned}
f(n) &= O(f(n)) \\
c \times O(f(n)) &= O(f(n)) \qquad c = constante \\
O(f(n)) + O(f(n)) &= O(f(n)) \\
O(O(f(n))) &= O(f(n)) \\
O(f(n)) + O(g(n)) &= O(max(f(n), g(n))) \\
O(f(n))O(g(n)) &= O(f(n)g(n)) \\
f(n)O(g(n)) &= O(f(n)g(n))
\end{aligned}
$$

Notação Assintótica

operações consecutivas - some seus tempos:

```
for ( int x = 1; x < N; x++ )
  <operação qualquer>;
for ( int x = 1; x < N; x++ )
  for ( int y = 1; y < N; y++ )
    <operação qualquer>;
```

tempo total?

Notação Assintótica

operações consecutivas - some seus tempos:

```
for ( int x = 1; x < N; x++ )
  <operação qualquer>;
for ( int x = 1; x < N; x++ )
  for ( int y = 1; y < N; y++ )
    <operação qualquer>;
```

tempo total?

$$\boxed{T(N) = N + N^2}$$

Notação Assintótica

operações consecutivas - some seus tempos:

```
for ( int x = 1; x < N; x++ )
  <operação qualquer>;
for ( int x = 1; x < N; x++ )
  for ( int y = 1; y < N; y++ )
    <operação qualquer>;
```

tempo total?

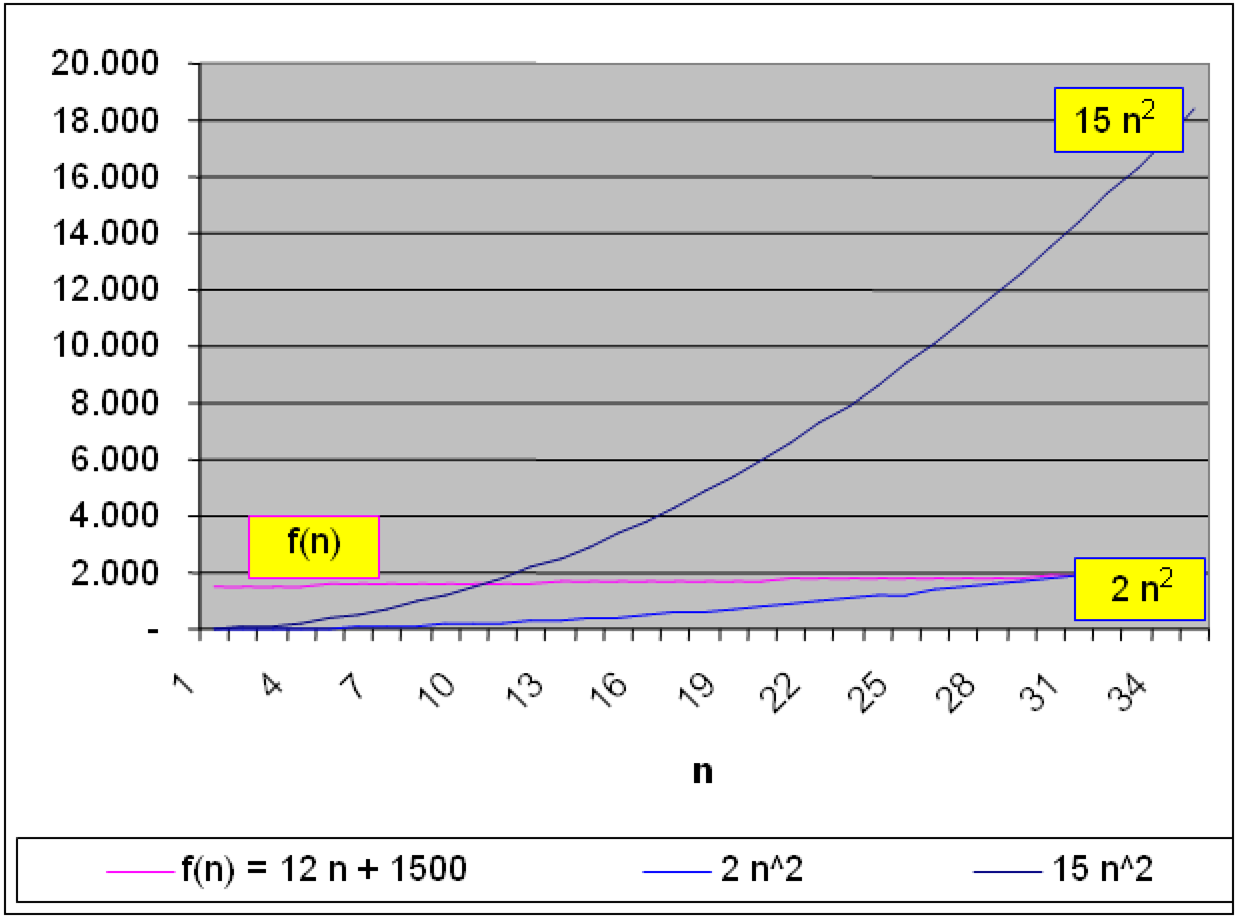
$T(N) = N + N^2$
$T(N) = O(N^2)$
$N + N^2 = O(N^2)$

Notação Assintótica

$$f(n) = 12n + 1500 \quad \text{é} \quad O(n^2)?$$

Notação Assintótica

$f(n) = 12n + 1500$ é $O(n^2)$?

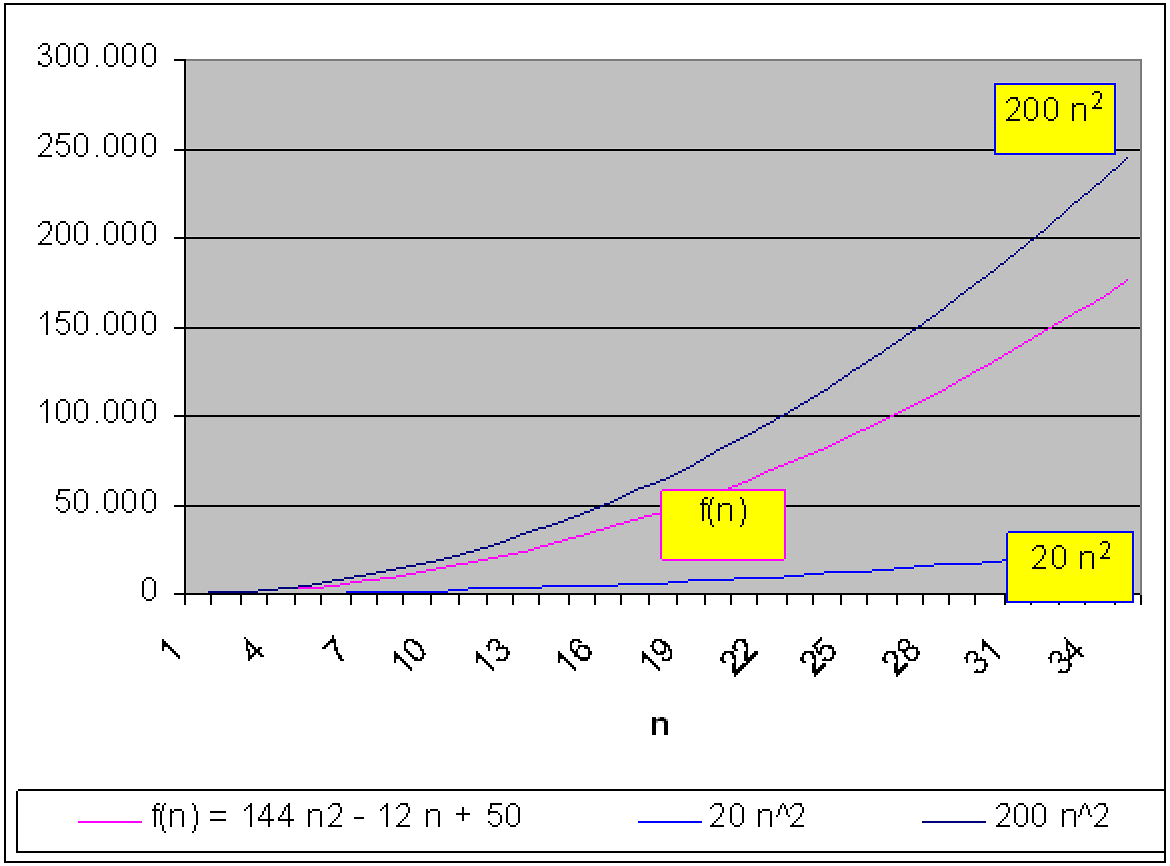


Notação Assintótica

$f(n) = 144\, n^2 - 12\, n + 50$ é $O(n^2)$?

Notação Assintótica

$f(n) = 144\ n^2 - 12\ n + 50$ é $O(n^2)$?

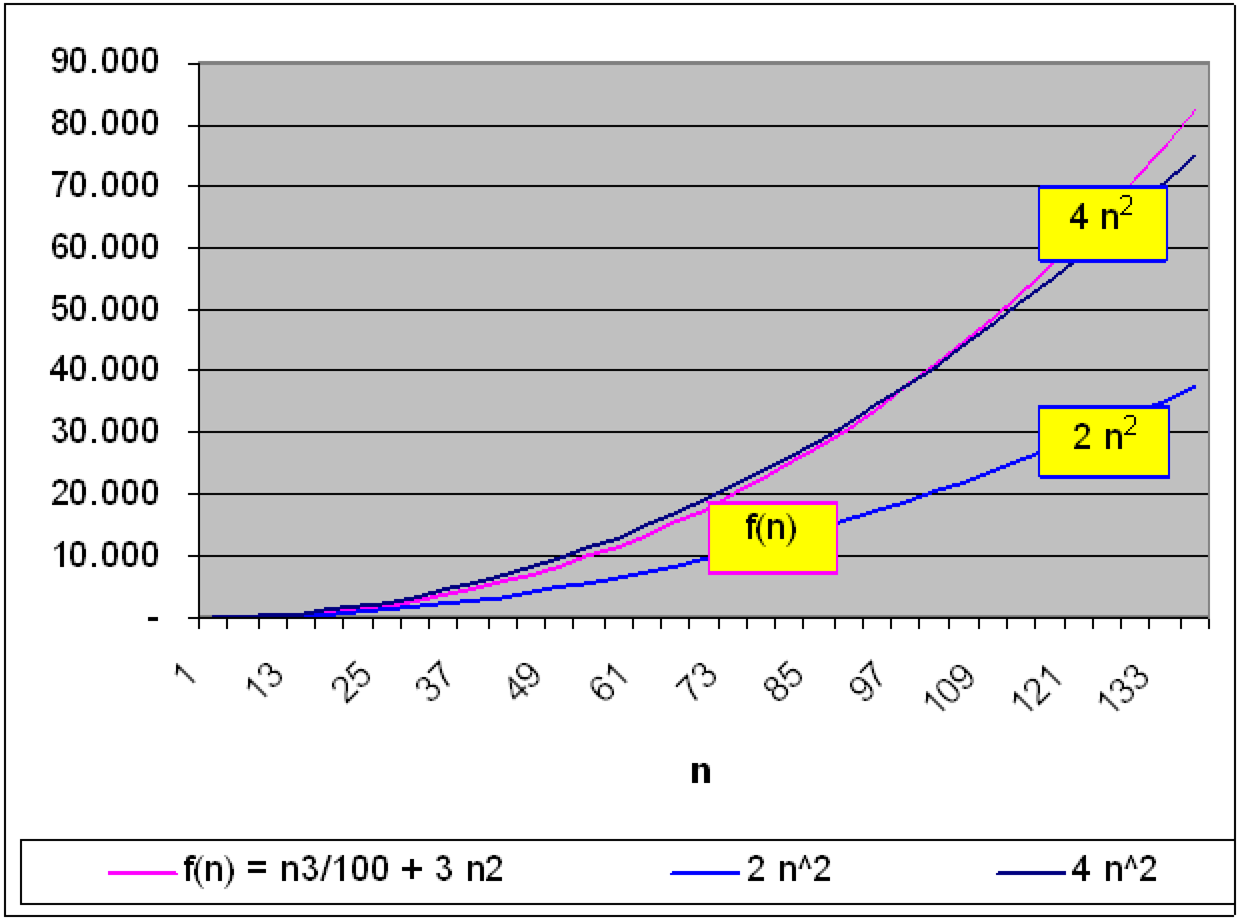


Notação Assintótica

$$f(n) = n^3/100 + 3\,n^2 \text{ é } O(n^2)?$$

Notação Assintótica

$f(n) = n^3/100 + 3\ n^2$ é $O(n^2)$?



Notação Assintótica

- Verifique as afirmações

$6n^4 + 12\,n^3 + 12$ é $O(n^4)$, $O(n^5)$ e $O(n^6)$

$3n^2 + 12\,n \log n$ é $O(n^2)$, $O(n^5)$ e $O(n)$

$5n^2 + n\,(\log n) + 12$ é $O(n^2)$ e $O(n^5)$

$\log n + 4$ é $O(\log n)$ e $O(n)$

Notação Assintótica

$6n^4 + 12\,n^3 + 12$	$\in O(n^4)$	$\in O(n^5)$	$\notin O(n^3)$
$3n^2 + 12\,n\log n$	$\in O(n^2)$	$\in O(n^5)$	$\notin O(n)$
$5n^2 + n\,(\log n) + 12$	$\in O(n^2)$	$\in O(n^5)$	
$\log n + 4$	$\in O(\log n)$	$\in O(n)$	

Notação Assintótica

- Dadas as funções de crescimento f(n) e g(n)

	$f(n)$ é $O(g(n))$?	$g(n)$ é $O(f(n))$?
$g(n)$ cresce mais	Sim	Não
$f(n)$ cresce mais	Não	Sim
Mesma taxa	Sim	Sim

Muita atenção

- O sinal de igualdade aqui não tem o significado habitual!!!

$$10\,n^2 + 10 \log n = O(n^2)$$

$$2\,n^2 - 3 = O(n^2)$$

$$\cancel{10\,n^2 + 10 \log n = 2\,n^2 - 3}$$

Referências de Material

- Adaptado do material de
  - Professor Alessandro L. Koerich da *Pontifícia Universidade Católica do Paraná (PUCPR)*
  - Professor Humberto Brandão da *Universidade Federal de Alfenas (Unifal-MG)*
  - Professor Ricardo Linden da *Faculdade Salesiana Maria Auxiliadora (FSMA)*
  - Professor Antonio Alfredo Ferreira Loureiro da *Universidade Federal de Minas Gerais (UFMG)*

Referências Bibliográficas

- CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; (2002). Algoritmos –Teoria e Prática. Tradução da 2ª edição americana. Rio de Janeiro. Editora Campus
- TAMASSIA, ROBERTO; GOODRICH, MICHAEL T. (2004). Projeto de Algoritmos -Fundamentos, Análise e Exemplos da Internet
- ZIVIANI, N. (2007). Projeto e Algoritmos com implementações em Java e C++. São Paulo. Editora Thomson
- http://www.ime.usp.br/~pf/analise_de_algoritmos/aulas/Oh.html