

#### Complexidade de Algoritmos

Prof. Diego Buchinger diego.buchinger@outlook.com diego.buchinger@udesc.br

Prof. Cristiano Damiani Vasconcellos cristiano.vasconcellos@udesc.br



#### Análise de Algoritmos

Analisar um algoritmo significa prever os recursos que algoritmo necessita. Por exemplo, memória, largura de banda e mais frequentemente o tempo de computação.

Para analisar um algoritmo é necessário definir um modelo de computação. O modelo de computação do computador tradicional é o RAM (*Random Access Machine*) onde as instruções são executadas em sequência, sem concorrência, e os dados são armazenados em células de memória com acesso aleatório.



#### Análise de Algoritmos

Contar o número de todas as instruções que são executadas pelo algoritmo.

Por exemplo: m load, n store, o add, p sub, q div, r mul, s call, t ret, u cmp, v jump, etc.



O número de instruções e o tempo de execução depende do processador, compilador, velocidade de acesso à memória, tamanho de memória (cache e ram) etc.



Comparação de desempenho na resolução de sistemas lineares considerando tempos de operações de um computador real:

| n  | Método de Cramer             | Método de Gauss |
|----|------------------------------|-----------------|
| 2  | 22 µs                        | 50 µs           |
| 3  | 100 µs                       | 159 µs          |
| 4  | 463 µs                       | 353 µs          |
| 5  | 2,15 ms                      | 666 µs          |
| 10 | 4,62 s                       | 4,95 ms         |
| 20 | 247 dias                     | 38,63 ms        |
| 40 | 1,45 * 10 <sup>13</sup> anos | 0,315 s         |



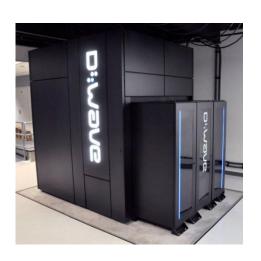
• Ok, mas e o avanço tecnológico, produzindo máquinas cada vez mais rápidas não faz o estudo de complexidade perder importância?



100x mais rápido



2<sup>b</sup> mais rápido



Computador 19xx

Computador 2016

Computador quântico



• Análise de impacto do aumento de velocidade dos computadores para o Método de Cramer:

| n  | Computador 19xx              | Computador 2016              |
|----|------------------------------|------------------------------|
| 3  | 100 µs                       | 1 µs                         |
| 5  | 2,15 ms                      | 21,5 µs                      |
| 7  | 46,274 ms                    | 463 µs                       |
| 10 | 4,62 s                       | 46,2 ms                      |
| 12 | 1,66 min                     | 1 s                          |
| 15 | 2,76 horas                   | 1,656 min                    |
| 20 | 247 dias                     | 2,47 dias                    |
| 40 | 1,45 * 10 <sup>13</sup> anos | 1,45 * 10 <sup>11</sup> anos |



#### Análise de Algoritmos

Prever os recursos de que o algoritmo necessitará

A complexidade vem ganhando destaque a ponto de que alguns autores dizem que este tema é o coração da Computação [Toscani e Veloso, 2001].

- Complexidade na fase de projeto do algoritmo
- Intratabilidade de problemas:
  - Problemas NP-Completos e NP-Difícil
  - Soluções alternativas (aproximações), uso de programação dinâmica.



#### Programa e Plano de Ensino

- Plano de Ensino
  - Objetivos e ementa
  - Conteúdo programático
  - Avaliação
  - Bibliografia
- Plano de Aulas

Disponível na página!



### Bibliografia

Algoritmos. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Cliford Stein. Campus. [**Bíblia**]

Algorithms. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani. McGraw Hill.

#### **Complementar:**

Complexidade de Algoritmos. Toscani, L.V. e Veloso, P.A.S. Instituto de Informática da UFRGS. Editora Sagra.



#### Atividade 1

- Elabore o melhor algoritmo para receber uma sequencia de 'n' números inteiros e dizer quantas vezes o número 'm' apareceu nesta sequência.
- NOTA: existe alguma consideração diferente caso 'm' seja um inteiro entre 0 e 10.000, ou um inteiro entre 0 e 1.000.000.000.000?