Disciplinas CCM-001 / MCTA003-17
Análise de Algoritmos (e Estruturas de Dados)

Revisão de conceitos importantes

Profa. Carla Negri Lintzmayer

carla.negri@ufabc.edu.br

www.professor.ufabc.edu.br/∼carla.negri

Centro de Matemática, Computação e Cognição – Universidade Federal do ABC

UFABC

Conteúdo

Estes slides contêm um resumo de conceitos matemáticos e de computação que são utilizados no decorrer da disciplina.

Revisão de conceitos matemáticos

Conceitos matemáticos

- Seção 1: Conjuntos
- Seção 2: Potenciação
- Seção 3: Logaritmos
- Seção 4: Polinômios e expressões racionais
- Seção 5: Funções
- Seção 6: Somatórios
- Seção 7: Equações e inequações
- Seção 8: Métodos de demonstração

Materiais extras:

- livro da disciplina de Bases Matemáticas
- vídeos de prova por indução

Conjuntos

Conjuntos

Conjunto

Um **conjunto** é um grupo de objetos representados como uma unidade.

Um conjunto podem conter qualquer tipo de objeto, incluindo números, símbolos, outros conjuntos, etc...

Elemento ou Membro

Os objetos em um conjunto são chamados de **elementos** ou **membros**.

Definindo conjuntos: listagem

Podemos definir um conjunto através da listagem de seus elementos, separados por vírgula, entre chaves.

Exemplos

$$\{1, 2, 3, 4\}$$

{banana, uva, 😆}

{🍇, 🍉, 🍍, 🍒, 🍌}

{😀, 😁, 😆, 😅}

Em um conjunto, a ordem em que listamos os elementos e elementos repetidos são irrelevantes.

$$\{1, 2, 3, 4\} = \{2, 1, 4, 3\} = \{1, 2, 2, 4, 3, 3\}$$

Pertinência em conjuntos

Se x é um membro do conjunto A, escrevemos $x \in A$, que pode ser lido como:

- x é um elemento de A,
- x é um membro de A,
- x pertence a A, ou
- x está em A.

Se $x \in A$, também dizemos que o conjunto A contém x

Não pertinência em conjuntos

Se x não é um membro do conjunto A, escrevemos $x \notin A$, que pode ser lido como:

- x não é um elemento de A
- x não é um membro de A
- x não pertence a A
- x não está em A

Se $x \notin A$, também dizemos que A não contém o elemento x

Nomeando conjuntos

Para não replicarmos a definição do conjunto toda vez que quisermos referenciá-lo, é comum nomeá-lo. Fazemos isso indicando que um conjunto é igual a um rótulo (variável).

Exemplo

$$A = \{1, 2, 3, 4\}$$

$$B = \{\text{banana, maçã, pera}\}$$

$$C = \{\text{😃, 😁, 😆, 😄}\}$$

Exemplo de uso

Seja $A = \{1, 2, 3, 4\}$ .... Como $2 \in A$ ...

Por convenção, rotulamos os conjuntos com letras maiúsculas.

Usando elipses na definição de conjuntos

Quando temos muitos elementos para listar na definição do conjunto, usamos elipses ("...") para denotar que a listagem de elementos continua de acordo com o padrão exibido.

Exemplos

$$\{1, 2, 3, \ldots, 100\}$$

$$\{0, 2, 4, \ldots, 100\}$$

$$\{x_1, x_2, \ldots, x_{89}\}$$

$$\{y_1, y_2, \ldots, y_k\}, \text{ onde } k \text{ é um inteiro}$$

Usando elipses na definição de conjuntos (cont.)

Na ausência de um padrão claro, assumimos que o incremento ocorre de um em um.

Exemplos

$$\{1, \ldots, 100\}$$

$$\{a_1, \ldots, a_{89}\}$$

Denotando conjuntos infinitos

Um conjunto X é **finito** se ele contém um número finito de membros.

Caso contrário, dizemos que X é um conjunto **infinito**.

Para denotar um conjunto **infinito** podemos usar elipses.

Exemplos

$$\{1, 2, 3, \ldots\}$$

$$\{1, 3, 5, \ldots\}$$

$$\{\ldots, -2, -1\}$$

$$\{\ldots, -2, -1, 0, 1, 2, \ldots\}$$

Cardinalidade

A **cardinalidade** ou **tamanho** de um conjunto X, denotado por $|X|$, é o número de elementos de X.

Exemplo

Se $A = \{0, 2, 4, 6\}$, $B = \{1, 2, \ldots, 100\}$, $C = \{1, 2, \ldots\}$, $D = \{\}$, então

$$|A| = 4 \qquad |B| = 100 \qquad |C| = \aleph_0 \qquad |D| = 0$$

Conjunto vazio

O conjunto $\{\}$, i.e., o conjunto que contém zero elementos, é chamado de conjunto vazio.

Ele é mais comumente denotado por $\emptyset$.

Conjuntos importantes

- $\emptyset = \{\}$ Conjunto vazio
- $\mathbb{N} = \{1, 2, \ldots\}$ Conjunto dos números naturais
- $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$ Conjunto dos números inteiros
- $\mathbb{Q}$ Conjunto dos números racionais
- $\mathbb{R}$ Conjunto dos números reais

Definindo conjuntos textualmente

Podemos definir conjuntos textualmente.

Exemplos

- "Seja X o conjunto de todos os números naturais menores ou iguais a 100"
- $X = \{$números naturais menores ou iguais a $100\}$

- Não é muito preciso, o que pode levar a ambiguidade na definição do conjunto.

Definindo conjuntos com predicados

Dado um predicado $P(x)$, podemos definir um conjunto com a notação

$$\{x\colon P(x)\} \qquad \text{ou} \qquad \{x \mid P(x)\}$$

- Os símbolos "$:$" e "$|$" são lidos como "tal que".
- Lê-se: "o conjunto formado pelos elementos x tais que $P(x)$ é verdadeiro".
- Todos os valores de x para os quais o predicado vale (é verdadeiro) pertencem ao conjunto que está sendo definido.
- Todos os valores de x para os quais o predicado não vale não pertencem ao conjunto.

O domínio do predicado pode aparecer no lado esquerdo:

$$\{x \in X\colon f(x)\} \qquad \text{ou} \qquad \{x \in X \mid f(x)\}$$

Definindo conjuntos com predicados (cont.)

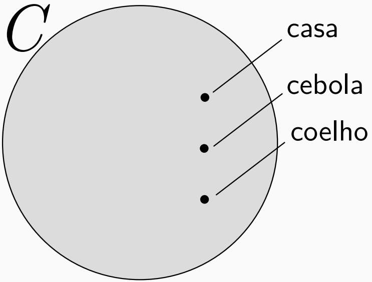
Exemplos

- $\{x\colon x$ é par$\}$
- $\{x \in \mathbb{N}\colon x$ é par$\} = \{0, 2, 4, 6, \ldots\}$
- $\{x\colon x \in \mathbb{N}$ e x é par$\} = \{0, 2, 4, 6, \ldots\}$
- $\{x\colon x = 0 \pmod 2\}$
- $\{x\colon x = 2k,$ para algum $k \in Z\}$
- $\{x \in \mathbb{Z}\} = \mathbb{Z}$
- $\{x \in \mathbb{Z}\colon 1 \leq x \leq 100\} = \{1, \ldots, 100\}$
- $\{(x, y)\colon 1 \leq x \leq 100$ e $1 \leq y \leq 10\} = \{(1, 1), (1, 2), \ldots, (1, 10), (2, 1), (2, 2), \ldots, (2, 10), \ldots, (100, 1), (100, 2), \ldots, (100, 10)\}$
- $\{x \in \mathbb{Z}\colon 1 \leq x \leq 100$ e x é múltiplo de $3\} = \{3, 6, 9, \ldots, 99\}$
- $\{x\colon x$ é um modelo de carro$\}$
- $\{x\colon x$ atuou com Chuck Norris e nasceu antes de $1980\}$

Desenhando conjuntos: diagrama de Venn

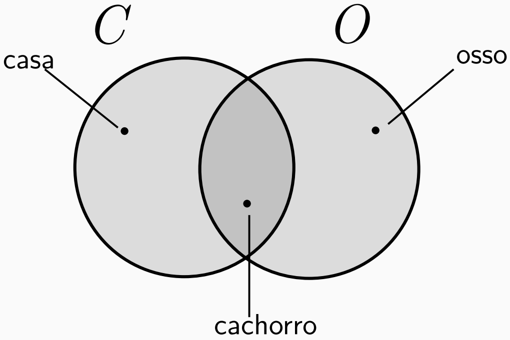
Diagrama de Venn é uma forma de representação gráfica para conjuntos.

- Conjuntos são regiões delimitadas por linhas circulares ou elípticas.
- (Alguns) membros podem ser representados por pontos.



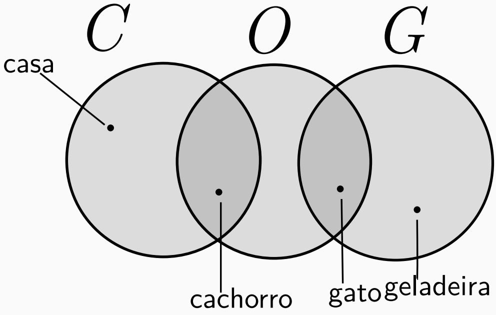
Desenhando conjuntos: diagrama de Venn (cont.)

- $C = \{$palavras que começam com a letra "c"$\}$
- $O = \{$palavras que terminam com a letra "o"$\}$



Desenhando conjuntos: diagrama de Venn (cont.)

- $C = \{$palavras que começam com a letra "c"$\}$
- $O = \{$palavras que terminam com a letra "o"$\}$
- $G = \{$palavras que começam com a letra "g"$\}$



Subconjunto

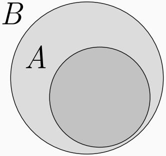
Subconjunto

Um conjunto A é um subconjunto de um conjunto B se para todo $x \in A$, temos que $x \in B$.

Notação: $A \subseteq B$

Se $A \subseteq B$, então B é um superconjunto de A.

Notação: $B \supseteq A$



Atenção!

$\emptyset \subseteq X$ para qualquer conjunto X.

Subconjunto Próprio

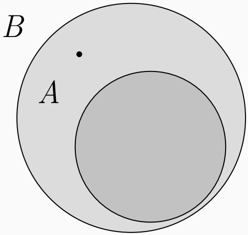
Subconjunto próprio

Um subconjunto A de um conjunto B é próprio se $A \neq B$.

Notação: $A \subsetneq B$ ou $A \subset B$.

Se $A \subset B$, então B é superconjunto próprio de A.

Notação: $B \supsetneq A$ ou $B \supset A$.

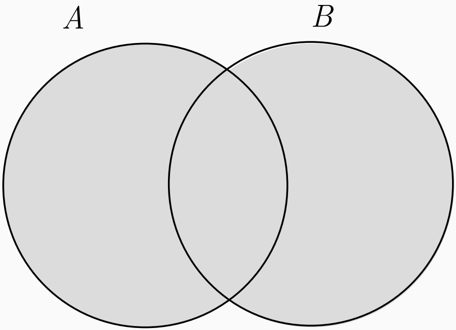


União

União

Dados dois conjuntos A e B quaisquer, sua união é o conjunto $A \cup B$ tal que

$$A \cup B = \{x \colon x \in A \text{ ou } x \in B\}$$

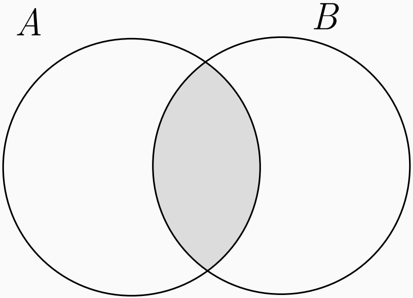


Interseção

Interseção

Dados dois conjuntos A e B quaisquer, sua interseção é o conjunto $A \cup B$ tal que

$$A \cap B = \{x \colon x \in A \text{ e } x \in B\}$$

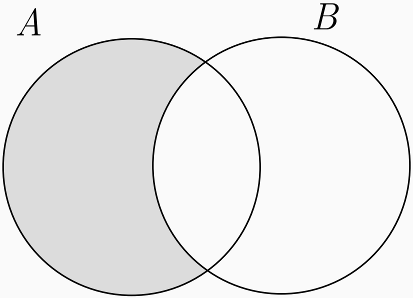


Diferença

Diferença

Dados dois conjuntos A e B quaisquer, sua diferença é o conjunto $A \setminus B$ tal que

$$A \setminus B = \{x \colon x \in A \text{ e } x \notin B\}$$



Conjunto potência

Conjunto potência

O conjunto potência de um conjunto A é o conjunto de todos os subconjuntos de A.

Notação: $\mathcal{P}(A)$.

Exemplo

Se $A = \{1, 2, 3\}$, então

$$\mathcal{P}(A) = \Big\{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}\Big\}$$

Produto cartesiano de dois conjuntos

Produto cartesiano

O produto cartesiano de dois conjuntos A e B é o conjunto $A \times B$ tal que

$$A \times B = \{(a, b) \colon a \in A \text{ e } b \in B\}$$

Exemplo

Sejam $A = \{1, 2\}$ e $B = \{a, b, c\}$.

$$A \times B = \big\{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c)\big\}$$

Produto cartesiano de k conjuntos

Produto cartesiano

O produto cartesiano dos conjuntos $A_1, A_2, \ldots, A_k$ é o conjunto $A_1 \times A_2 \times \cdots \times A_k$ tal que

$$A_1 \times A_2 \times \cdots \times A_k = \left\{(a_1, a_2, a_3, \ldots, a_k) \colon a_i \in A_i \text{ e } 1 \leq i \leq k\right\}$$

Produto cartesiano

O produto cartesiano do conjunto A com ele mesmo k vezes é denotado por A^k:

$$A^k = \underbrace{A \times A \times \cdots \times A}_{k \text{ termos}}$$

Potenciação

Potenciação

Potenciação

Seja $a \in \mathbb{R}$ e $x \in \mathbb{Z}$. A operação de potenciação é definida por

$$a^x = \begin{cases} a \times a \times a \times \cdots \times a \ (x \text{ vezes}) & \text{se } x > 0 \\ \dfrac{1}{a^x} & \text{se } x < 0 \\ 1 & \text{se } x = 0 \end{cases}$$

Exemplos

- $2^5 = 2 \times 2 \times 2 \times 2 \times 2 = 32$
- $4^{-2} = \dfrac{1}{4^2} = \dfrac{1}{16}$
- $2357^0 = 1$

Potenciação

Seja $a \in \mathbb{R}$ e $x = \frac{m}{n} \in \mathbb{Q}$. A operação de potenciação é definida por

$$a^x = a^{\frac{m}{n}} = (a^{\frac{1}{n}})^m$$

em que

$$a^{\frac{1}{n}} = b \quad \Leftrightarrow \quad a = b^n$$

$$a^{\frac{1}{x}} \text{ também é escrito } \sqrt[x]{a}$$

Assim,

$$a^{\frac{m}{n}} = (a^m)^{\frac{1}{n}} = \sqrt[n]{a^m}$$

Potenciação

Propriedades

- $a^x \times a^y = a^{x+y}$
- $\dfrac{a^x}{a^y} = a^{x-y}$
- $(a^x)^y = a^{xy}$
- $(a \times b)^x = a^x \times b^x$
- $\left(\dfrac{a}{b}\right)^x = \dfrac{a^x}{b^x}$

Logaritmos

Logaritmo

Sejam n e a números positivos, com $a \neq 1$.

O **logaritmo de n na base a**, denotado $\log_a n$, é o valor x tal que

- x é o expoente a que a deve ser elevado para produzir n; ou,
- x é a quantidade de vezes que a deve ser multiplicado por a para produzir n; ou,
- x é a quantidade de vezes que n deve ser dividido por a para produzir 1.

Assim, por definição,

$$\log_a n = b \text{ se e somente se } a^b = n$$

Vamos considerar que $\lg n = \log_2 n$ e $\log n = \log_{10} n$.

Logaritmo: exemplos

Exemplo

O logaritmo de 32 na base 2 é 5 ($\lg 32 = 5$), pois

- $2^5 = 32$
- $\frac{32}{2} = 16$, $\frac{16}{2} = 8$, $\frac{8}{2} = 4$, $\frac{4}{2} = 2$, $\frac{2}{2} = 1$ (5 divisões)

Exemplo

O logaritmo de 81 na base 3 é 4 ($\log_3 81 = 4$), pois

- $3^4 = 81$
- $\frac{81}{3} = 27$, $\frac{27}{3} = 9$, $\frac{9}{3} = 3$, $\frac{3}{3} = 1$ (4 divisões)

Logaritmo: como calcular?

Alguns logaritmos são fáceis de calcular:

- $\lg 1024 = 10$ (pois 1024 é potência de 2)
- $\log 100 = 2$ (pois 100 é potência de 10)
- $\log_5 125 = 3$ (pois 125 é potência de 5)

Outros precisam de calculadora:

- $\lg 37 = 5,209453366$
- $\log 687 = 2,836956737$
- $\log_5 341 = 3,623552317$

Nesse curso, não usaremos (e nem precisaremos de) calculadora.

Logaritmo: como calcular?

Podemos ter uma noção dos valores por meio de comparação com os logaritmos fáceis:

- $\lg 37$ é algum valor entre 5 e 6, pois
  - $\lg 32 < \lg 37 < \lg 64$
  - $\lg 32 = 5$ e $\lg 64 = 6$
- $\log 687$ é algum valor entre 2 e 3, pois
  - $\log 100 < \log 687 < \log 1000$
  - $\log 100 = 2$ e $\log 1000 = 3$
- $\log_5 341$ é algum valor entre 3 e 4, pois
  - $\log_5 125 < \log_5 341 < \log_5 625$
  - $\log_5 125 = 3$ e $\log_5 625 = 4$

Logaritmo: propriedades

Propriedades

- $\log_b 1 = 0$
- $\log_b b = 1$
- $b^{\log_b n} = n$
- $a^{\log_b n} = n^{\log_b a}$
- $\log_b b^n = n$
- $\log_b n = \dfrac{\log_a n}{\log_a b}$
- $\log_b(nm) = \log_b n + \log_b m$
- $\log_b \left(\frac{n}{m}\right) = \log_b n - \log_b m$
- $\log_b n^x = x \log_b n$

Logaritmo

Cuidado!

Note que

$$\log_b n^x \neq \log_b^x n\,,$$

pois

$$\log_b n^x = \log_b(n \times \cdots \times n) = \log_b n + \cdots + \log_b n = x \log_b n$$

e

$$\log_b^x n = (\log_b n)^x = (\log_b n) \times \cdots \times (\log_b n)$$

Exemplo

$\lg 32^3 = 3 \times \lg 32 = 15 = \lg 32768$
$\lg^3 32 = (\lg 32)^3 = 125$

Polinômios e expressões racionais

Polinômios

Polinômio

Sejam $k \in \mathbb{N}$ e $a_k, a_{k-1}, \ldots, a_1, a_0 \in \mathbb{R}$ constantes com $a_k \neq 0$.

Um **polinômio de grau** k **na variável** n é

$$p(n) = a_k n^k + a_{k-1} n^{k-1} + \cdots + a_1 n + a_0$$

Exemplos

- $3n^4 + 5n - 8$ $(k = 4,\ a_4 = 3,\ a_3 = a_2 = 0,\ a_1 = 5,\ a_0 = -8)$
- $-7n^9$ $(k = 9,\ a_9 = -7,\ a_8 = \cdots = a_0 = 0)$
- $\frac{2}{5}n^2 - 675n$ $(k = 2,\ a_2 = \frac{2}{5},\ a_1 = -675,\ a_0 = 0)$

Polinômios especiais

- Constante: de grau zero (39, -103, $\frac{56}{7}$, $\sqrt{98}$)
- Linear: de grau um ($34n + 8$, $-9n - 10$, $2n$, $\frac{1}{2}n - 98$)
- Quadrático: de grau dois ($n^2 + 7n$, $2n^2 - 8$, $3n^2 + n - 9$)

Expressões racionais

Expressão racional

Uma expressão racional é o quociente entre dois polinômios $p(n)$ e $q(n)$:

$$\frac{p(n)}{q(n)}$$

Propriedades

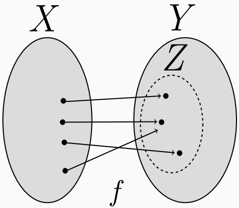
Se a, b, c e d são polinômios, então:

- $\dfrac{a}{b} + \dfrac{c}{d} = \dfrac{ad + bc}{bd}$
- $\dfrac{a}{b} \times \dfrac{c}{d} = \dfrac{ac}{bd}$
- $\dfrac{a/b}{c/d} = \dfrac{a}{b} \times \dfrac{d}{c} = \dfrac{ad}{bc}$

Funções

Função

Uma **função** é uma relação entre dois conjuntos que associa cada elemento do primeiro a exatamente um elemento do segundo.



Na função f da figura acima:

- X é o domínio da função f
- Y é o contradomínio da função f
- Z é a imagem da função f

Funções: notação

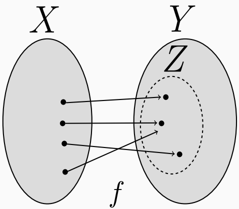
Notação

Se f é uma função com domínio X e contradomínio Y, então escrevemos:

$$f\colon X \to Y$$

Assim,

$$f(x) = y \qquad \Rightarrow \qquad x \in X \text{ e } y \in Y$$



Funções: representações

Seja f: $\{1, 2, 3\} \to \{2, 4, 6\}$.

Podemos representar f por uma expressão: $f(x) = 2x$ para $x \in \{1, 2, 3\}$.

Ou de forma tabular:

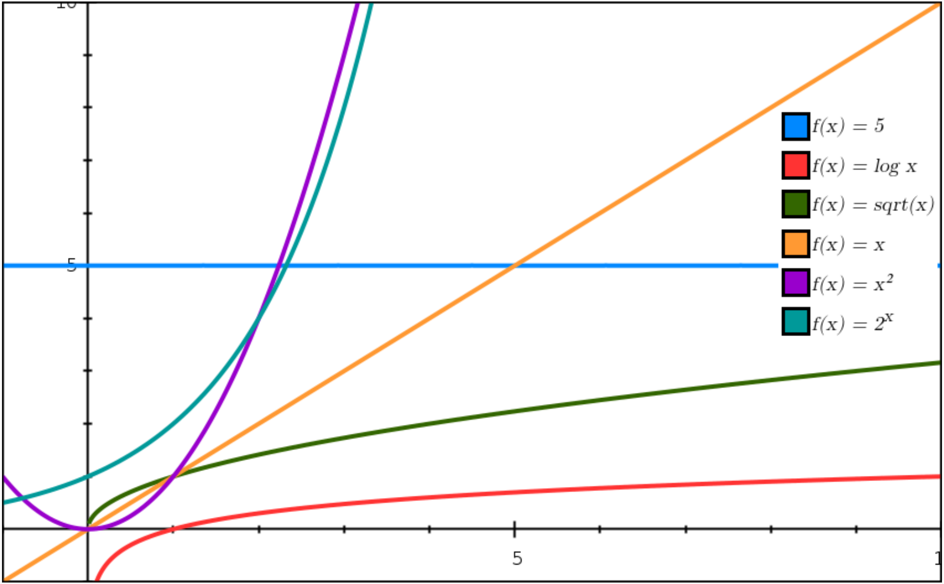
x	$f(x)$
1	2
2	4
3	6

Funções especiais

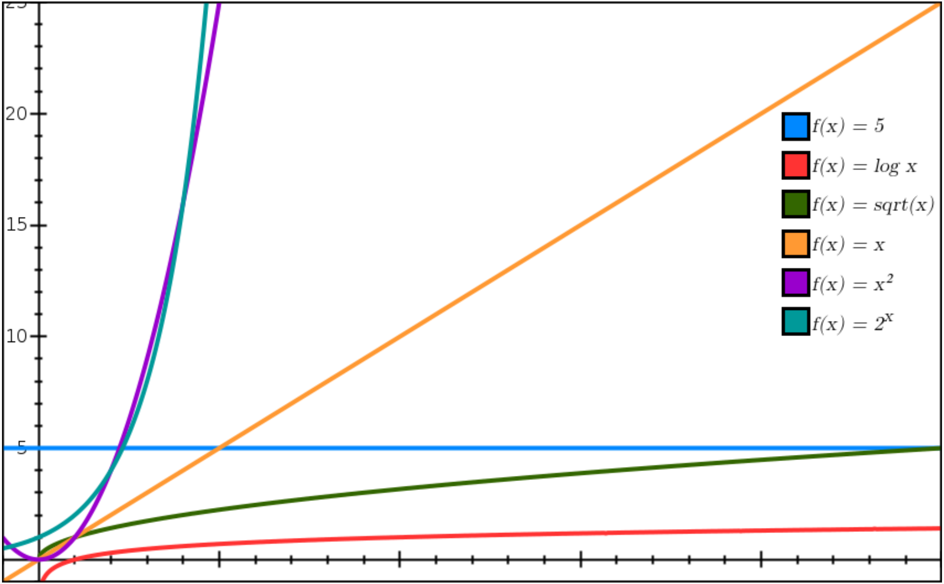
Sejam a, b e a_i, para $0 \leq i \leq k$, constantes:

- Constantes: $f(n) = a$
- Lineares: $f(n) = an + b$
- Polinomiais: $f(n) = a_k n^k + a_{k-1} n^{k-1} + \cdots + a_1 n + a_0$
- Racionais: $f(n) = p(n)/q(n)$
- Exponenciais: $f(n) = a^n$
- Logaritmicas: $f(n) = \log_a n$

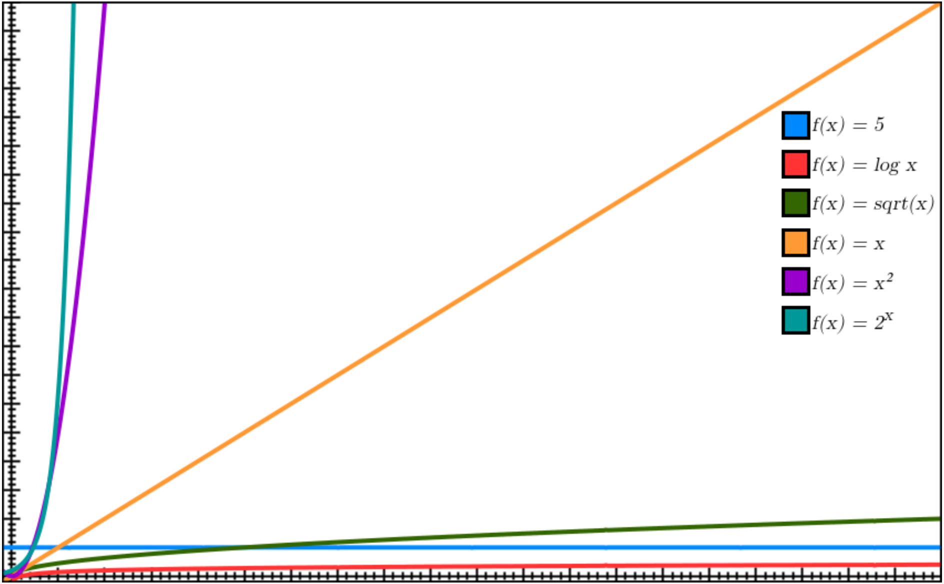
Funções



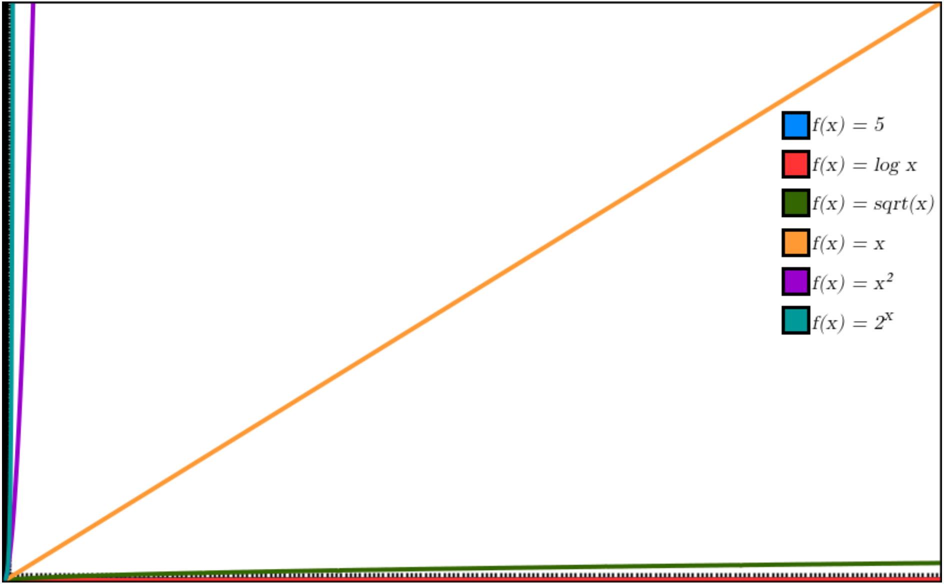
Funções



Funções



Funções



Piso e teto

Função piso

A função **piso** recebe um número real x e devolve o *maior inteiro que é menor ou igual a* x.

Notação: $piso(x) = \lfloor x \rfloor$

Exemplos

- $\lfloor 3,4 \rfloor = 3$
- $\lfloor 3 \rfloor = 3$
- $\lfloor 2,999998 \rfloor = 2$

Piso e teto

Função teto

A função **teto** recebe um número real x e devolve o *menor inteiro que é maior ou igual a* x.

Notação: $teto(x) = \lceil x \rceil$

Exemplos

- $\lceil 3,4 \rceil = 4$
- $\lceil 3 \rceil = 3$
- $\lceil 2,999998 \rceil = 3$

Piso e teto: propriedades

- $x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1$
- $\lfloor x \rfloor \leq \lceil x \rceil$
- $\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil = n$, se $n \in \mathbb{Z}$

Somatórios

Somatórios

Somatório

Um **somatório** é a adição de uma sequência de elementos que podem ser somados (números, funções, matrizes, polinômios).
Se $t_1, t_2, \ldots, t_n$ são n objetos que desejam ser somados, sua soma

$$t_1 + t_2 + \cdots + t_n$$

pode ser representada de forma compacta utilizando o símbolo $\sum$:

$$\sum_{i=1}^{n} t_i$$

onde i é o **índice do somatório**.

Os limites inferior e superior do índice do somatório não precisam ser sempre 1 e um valor n desconhecido.

Somatórios

Exemplos

- $$\sum_{i=1}^{8} i = 1+2+3+4+5+6+7+8$$
- $$\sum_{j=7}^{11} 4^j = 4^7 + 4^8 + 4^9 + 4^{10} + 4^{11}$$
- $$\begin{aligned}\sum_{k=0}^{4}(2k+1) &= (2\times 0+1)+(2\times 1+1)+\cdots+(2\times 4+1)\\ &= 1+3+5+7+9\end{aligned}$$
- $$\sum_{i=4}^{n} i3^i = (4\times 3^4)+(5\times 3^5)+\cdots+(n\times 3^n)$$
- $$\sum_{k=0}^{n} k^{k+1} = 0^1+1^2+2^3+3^4+\cdots+n^{n+1}$$

Somatórios: propriedades

Sejam $x, y \in \mathbb{Z}$ com $x \leq y$ e $c \in \mathbb{R}$ constante:

- $\displaystyle\sum_{i=x}^{y} ca_i = c\sum_{i=x}^{y} a_i$
- $\displaystyle\sum_{i=x}^{y}(a_i \pm b_i) = \sum_{i=x}^{y} a_i \pm \sum_{i=x}^{y} b_i$
- $\displaystyle\sum_{i=x}^{y} a_i = \sum_{i=x}^{z} a_i + \sum_{i=z+1}^{y} a_i$, com $x \leq z \leq y$

Somatórios: notações

Outras notações possíveis:

- $\displaystyle\sum_{3 \leq k < 8} a_k = a_3 + a_4 + \cdots + a_7 = \sum_{k=3}^{7} a_k$
- $\displaystyle\sum_{x \in S}$ indica que a soma terá um termo para cada elemento x contido em um conjunto S
  - se $S = \{4, 8, 13\}$, então $\displaystyle\sum_{x \in S} x = 4 + 8 + 13$

Somatórios

Alguns somatórios especiais:

- $\displaystyle\sum_{i=x}^{y} c = (y - x + 1)c$
- $\displaystyle\sum_{i=1}^{n} i = 1 + 2 + \cdots + n = \frac{n(n+1)}{2}$
- $\displaystyle\sum_{i=0}^{n} c^i = \frac{1 - c^{n+1}}{1 - c} = \frac{c^{n+1} - 1}{c - 1}$

Equações e inequações

Equações

Equação

Uma **equação** é uma proposição sobre a igualdade de duas expressões que têm o mesmo domínio:

$$f(n) = g(n)$$

Exemplos

- $2x^2 + 6x - 9 = 0$
- $34x = -9x$
- $3x^2 - 10 = 2x + 18$

Equações: propriedades

- se $f(n) = g(n)$, então $f(n) + c = g(n) + c$
- se $f(n) = g(n)$ e $c \neq 0$, então $cf(n) = cg(n)$

Inequações

Inequação

Uma **inequação** é uma proposição sobre duas expressões separadas por um símbolo de desigualdade:

$$f(n) < g(n) \quad f(n) \leq g(n) \quad f(n) > g(n) \quad f(n) \geq g(n)$$

Exemplos

- $2x^2 + 6x - 9 < 0$
- $34x \geq -9x$
- $3x^2 - 10 \leq 2x + 18$

Inequações: propriedades

- $a \leq a + b$, com $b \geq 0$.
- $a \geq a - b$, com $b \geq 0$.
- se $a \leq b$, então $a + c \leq b + c$.
- se $a \leq b$ e $b \leq c$, então $a \leq c$.
- se $a \leq b$ e $c \leq d$, então $a + c \leq b + d$.
- se $a \leq b$ e $c \geq 0$, então $ac \leq bc$.
- se $a \leq b$ e $c \leq 0$, então $ac \geq bc$.
- se $a \leq b$, então $-a \geq -b$.
- se $a \leq b$ e a e b têm mesmo sinal, então $\frac{1}{a} \geq \frac{1}{b}$.
- se $a < 0 < b$, então $\frac{1}{a} < \frac{1}{b}$.

Métodos de demonstração

Método dedutivo

- Como ter certeza que nossa resposta é correta?
- Como transmitir aos outros essa certeza?
- Começamos por *axiomas*: fatos simples que todos concordam que são verdade.
- Desenvolvemos um raciocínio a partir deles usando *regras de inferência*.

Se você encontrou um **contraexemplo** para sua questão, pode ter certeza de que ela está incorreta.

Suponha que $x > 3$. Então $x^2 - 2y > 5$.

Essa afirmação é falsa.

Tome, por exemplo, $x = 4$ e $y = 9$.

Nesse caso, $x^2 - 2y = 16 - 18 = -2 \not> 5$. C.Q.D.

Método dedutivo

Porém, a única forma de ter certeza que sua questão está correta é **demonstrando-a** para qualquer valor.

Suponha que $x > 3$ e $y < 2$. Então $x^2 - 2y > 5$.

Como $x > 3$, temos que $x^2 > 9$.

Como $y < 2$, temos que $-y > -2$ e $-2y > -4$.

Assim, $x^2 - 2y > 9 - 4 = 5$. C.Q.D.

Demonstração

Demonstração

Uma **demonstração** é uma argumentação precisa que procura convencer o leitor de que uma certa proposição, previamente enunciada, está correta.
É uma sequência de afirmações organizada de tal forma que:

- cada afirmação é consequência simples das afirmações anteriores e das hipóteses da proposição em discussão;
- a última afirmação é a proposição que se deseja demonstrar;
- um símbolo determina seu término ($\square$ ou C.Q.D).

Infelizmente, uma demonstração apenas descreve os passos necessários para chegar à conclusão, sem explicar o raciocínio utilizado (que nem sempre é óbvio).

Demonstração

- Infelizmente não existe um "algoritmo" para escrever demonstrações.
- Em geral também não se consegue escrever uma demonstração de imediato.
- Por outro lado, existem algumas técnicas gerais.
- Qual técnica usar?
  - A que funcione
  - Talvez tenhamos que recomeçar várias vezes, mas com o tempo ganhamos alguma intuição.
- Algumas dicas gerais também são úteis:
  - Entenda bem o que deve ser provado.
  - Tenha paciência.
  - Seja conciso.
  - Se você não sabe por onde começar, escreva todas as definições dos termos que estão nas hipóteses.

Demonstração direta

Argumente a veracidade da proposição diretamente.

Teorema

Sejam m e n números inteiros. Se m e n são pares, então $m + n$ é par.

Demonstração.

Sejam m e n inteiros pares. Como m é par, existe inteiro r tal que $m = 2r$. Similarmente, existe inteiro s tal que $n = 2s$. Portanto, $m + n = 2r + 2s = 2(r + s)$. Como $r + s$ é inteiro, temos que $m + n$ é par. C.Q.D.

Demonstração por contradição

Assuma que o que você quer provar é falso e construa argumentos até chegar a um absurdo.

Teorema

Sejam m e n números inteiros. Se m e n são pares, então $m + n$ é par.

Demonstração.

Sejam m e n inteiros pares. Como m é par, existe inteiro r tal que $m = 2r$. Similarmente, existe inteiro s tal que $n = 2s$. Assuma, para fins de contradição, que $m + n$ é ímpar. Então existe inteiro t tal que $m + n = 2t + 1$. Assim, $2r + 2s = 2t + 1$, ou seja, $2(r + s - t) = 1$, o que é uma contradição, pois $r + s - t$ é um inteiro e 1 é ímpar Então $m + n$ deve ser par. C.Q.D.

Demonstração por indução

- Se $n \in \mathbb{N}$, então $n^2 + n + 41$ é primo?
  - Vale para $n = 1, 2, \ldots, 39$ mas $40^2 + 40 + 41 = 41^2$, que não é primo.
- Se n é inteiro positivo, então $991n^2 + 1$ não é quadrado perfeito?
  - Não vale para $x = 12055735790331359447442538767$ mas vale para todos os números $n < x$.
- A soma dos n primeiros números ímpares é n^2?
  - Note que $1 = 1^2$, $1 + 3 = 2^2$, $1 + 3 + 5 = 3^2$, $1 + 3 + 5 + 7 = 4^2$ e $1 + 3 + 5 + 7 + 9 = 5^2$, mas é possível que seja apenas uma coincidência.

Indução

Considere a seguinte afirmação:

Para todo $n \in \mathbb{N}$, a soma $2^0 + 2^1 + \cdots + 2^n$ é igual a $2^{n+1} - 1$.

- Se $n = 0$, a soma é $2^0 = 1$ e $2^{0+1} - 1 = 2 - 1 = 1$ e a frase vale.
- Se $n = 1$, a soma é $2^0 + 2^1 = 1 + 2 = 3$ e $2^{1+1} - 1 = 4 - 1 = 3$ e a frase vale.
- Se $n = 2$, a soma é $2^0 + 2^1 + 2^2 = 1 + 2 + 4 = 7$ e $2^{2+1} - 1 = 8 - 1 = 7$ e a frase vale.
- Se $n = 3$, a soma é $2^0 + 2^1 + 2^2 + 2^3 = 1 + 2 + 4 + 8 = 15$ e $2^{3+1} - 1 = 16 - 1 = 15$ e a frase vale.

Provar a afirmação dessa forma não só é cansativo como é impossível e ainda pode ser inútil se ela for falsa!

Indução

Acontece que o objeto maior

$$2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7$$

contém objetos menores, como

$$(2^0 + 2^1 + 2^2 + 2^3 + 2^4) + 2^5 + 2^6 + 2^7$$

$$(2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6) + 2^7$$

E se soubermos provar a validade de um desses objetos menores?

$$2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 = 2^7 - 1$$

Então podemos usar isso para provar o maior:

$$(2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6) + 2^7 =$$

$$(2^7 - 1) + 2^7 = 2 \times 2^7 - 1 = 2^8 - 1\,.$$

Indução

Mas isso vale para qualquer valor de n (que não seja o 0):

$$2^0 + 2^1 + \cdots + 2^{n-1} + 2^n = \left(2^0 + 2^1 + \cdots + 2^{n-1}\right) + 2^n$$

E se a propriedade valer para $2^0 + 2^1 + \cdots + 2^{n-1}$, isto é, se $2^0 + 2^1 + \cdots + 2^{n-1} = 2^n - 1$, então substituindo acima:

$$\begin{aligned} 2^0 + 2^1 + \cdots + 2^{n-1} + 2^n &= \left(2^0 + 2^1 + \cdots + 2^{n-1}\right) + 2^n \\ &= \left(2^n - 1\right) + 2^n \\ &= 2 \times 2^n - 1 \\ &= 2^{n+1} - 1 \end{aligned}$$

Usamos um processo de redução: no objeto maior (n), enxergamos o objeto menor ($n-1$), supomos que o resultado valia para o menor e provamos que ele vale para o maior.

Indução

Podemos usar um processo de aumentação: supondo que o resultado no objeto melhor vale, partimos dele e *construímos* o resultado para o objeto maior:

$$
\begin{aligned}
2^0 + 2^1 + \cdots + 2^{n-1} &= 2^n - 1 \\
2^0 + 2^1 + \cdots + 2^{n-1} + 2^n &= 2^n - 1 + 2^n \\
2^0 + 2^1 + \cdots + 2^{n-1} + 2^n &= 2 \times 2^n - 1 \\
2^0 + 2^1 + \cdots + 2^{n-1} + 2^n &= 2^{n+1} - 1
\end{aligned}
$$

Questão de gosto

Veja que usei os termos $n-1$ e n, mas poderíamos usar n e $n+1$. Isso é apenas uma questão de nomenclatura e gosto pessoal.

Indução

Nos dois processos, estamos criando uma relação entre os objetos de interesse:

$$2^0 + \cdots + 2^n \rightsquigarrow 2^0 + \cdots + 2^n + 2^{n+1}$$

E há uma cadeia de relações, até que se chegue em um ou mais objetos de tamanho bem pequeno (nenhum objeto de tamanho menor do que esse pode ser resolvido):

$$2^0 \rightsquigarrow 2^0 + 2^1 \rightsquigarrow 2^0 + 2^1 + 2^2 \rightsquigarrow 2^0 + 2^1 + 2^2 + 2^3 \rightsquigarrow 2^0 + 2^1 + 2^2 + 2^3 + 2^4 \rightsquigarrow \cdots$$

No exemplo, a abordagem utilizada só funciona para $n > 0$, por isso quando $n = 0$ precisamos resolver diretamente.

Indução

Consiste em mostrar diretamente a proposição para alguns objetos (casos base) e mostrar uma relação entre um objeto qualquer e algum(ns) objeto(s) menor(es) (seja pelo processo de redução ou aumentação), justamente porque isso **implicitamente** constrói toda a cadeia.

Indução

Para provar que $P(n)$ é verdadeira $\forall n \in \mathbb{N}$, com $n \geq n_0$ para algum $n_0 \in \mathbb{N}$:

- Mostre que $P(n_0)$ é verdade (e talvez $P(n_0 + 1), \ldots, P(n_0 + k)$ também, para algum $k \geq 0$).
- Considere um $n > n_k$ qualquer.
- Assuma que $P(k)$ vale, para todo k tal que $n_0 \leq k < n$.
- Prove $P(n)$.

Indução: exemplo

Teorema

A soma dos n primeiros números naturais ímpares é n^2.

Demonstração.

Por indução em n.

Quando $n = 1$, o primeiro natural ímpar é 1, que é igual a 1^2.

Seja $n > 1$ um número natural qualquer. Suponha que a soma dos k primeiros naturais ímpares é k^2, para qualquer $1 \leq k < n$.

Vamos verificar se a soma dos n primeiros naturais ímpares $(1 + 3 + 5 + \cdots + (2n - 3) + (2n - 1))$ é n^2.

Note que $1 + 3 + 5 + \cdots + (2n - 3) = (n - 1)^2$, por hipótese de indução. Então

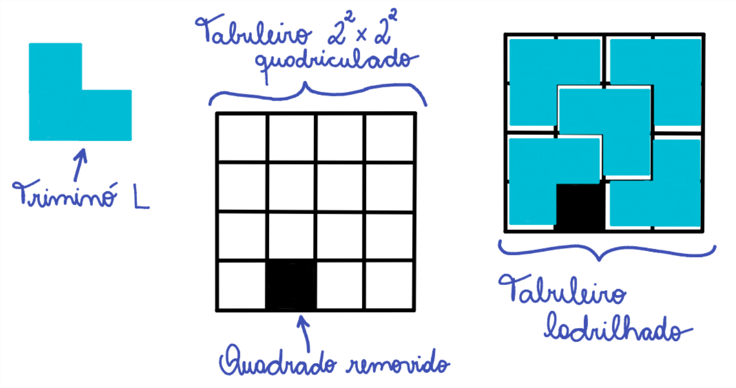
$$\begin{aligned} 1 + 3 + 5 + \cdots + (2n - 3) + (2n - 1) &= (n - 1)^2 + (2n - 1) \\ &= n^2 - 2n + 1 + 2n - 1 = n^2 \ . \end{aligned}$$

C.Q.D.

Indução: outro exemplo

Teorema

Seja n um inteiro positivo. Todo tabuleiro de damas de tamanho $2^n \times 2^n$ com um quadrado removido pode ser ladrilhado por triminós em forma de "L".

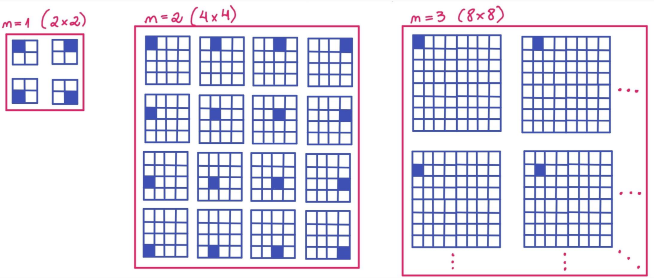


Indução: cuidados

Teorema

Seja n um inteiro positivo. Todo tabuleiro de damas de tamanho $2^n \times 2^n$ com um quadrado removido pode ser ladrilhado por triminós em forma de "L".

Observe como existem vários objetos para um mesmo valor n:



A demonstração deve valer para qualquer um desses objetos!

Indução: cuidados

- Quando existem vários objetos para um mesmo dado n, a abordagem por *aumentação* mencionada anteriormente precisa de mais cuidados do que a prova por *redução*.
- **Aumentação**: consideramos que temos algum objeto menor para o qual já vale a proposição e construímos algum objeto maior, de tamanho n, a partir dele.
  - É preciso ter certeza de que esse processo é geral o suficiente para criar **todos** os objetos de tamanho n.
- **Redução**: consideramos que temos algum objeto **qualquer** de tamanho n para o qual queremos provar a proposição e "removemos" algo dele para gerar um objeto de tamanho menor.
  - Quase sempre a remoção já cria um objeto menor válido, para o qual a hipótese vale diretamente.

Indução: cuidados

- Em qualquer abordagem: é preciso garantir que os objetos criados satisfazem todas as propriedades desejadas antes de assumir que o resultado vale.
  - No primeiro exemplo: o objeto é uma soma, mas ela precisa ter a propriedade de ser sobre as primeiras potências de 2.
  - No segundo exemplo: o objeto também é uma soma, mas ela precisa ter a propriedade de ser sobre os primeiros números naturais ímpares.
  - No terceiro exemplo: o objeto é um tabuleiro, mas ele precisa ser quadrado, com lados de tamanho iguais a potências de 2 e é necessário que um quadrado esteja removido.

Indução: voltando ao exemplo

Teorema

Seja n um inteiro positivo. Todo tabuleiro de damas de tamanho $2^n \times 2^n$ com um quadrado removido pode ser ladrilhado por triminós em forma de "L".

Indução: voltando ao exemplo

Teorema

Seja n um inteiro positivo. Todo tabuleiro de damas de tamanho $2^n \times 2^n$ com um quadrado removido pode ser ladrilhado por triminós em forma de "L".

Demonstração.

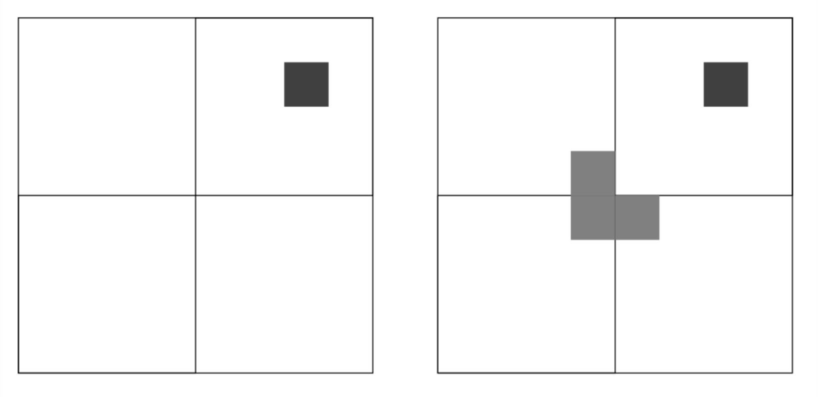
- Vamos provar por indução em n.
- Quando $n = 1$, um tabuleiro 2×2 certamente pode ser coberto por um triminó, independente de onde está o quadrado removido.
- Seja $n > 1$ um inteiro qualquer.
- Suponha que todo tabuleiro de tamanho $2^k \times 2^k$ com um quadrado removido pode ser ladrilhado por triminós, para $1 \leq k < n$.

Indução: exemplo (cont.)

- Considere agora um tabuleiro $2^n \times 2^n$ com algum quadrado removido.
  - Podemos dividi-lo em 4 subtabuleiros menores de tamanho $2^{n-1} \times 2^{n-1}$ cada (como $n > 1$, vale que $2^{n-1} < 2^n$).
  - Suponha, s.p.g., que o quadrado removido está no subtabuleiro superior direito.
  - Por hipótese, este pode ser ladrilhado.
  - Escolhemos quadrados específicos para remover nos outros três subtabuleiros (as casas centrais do tabuleiro original).
  - Agora podemos usar a hipótese e ladrilhá-los.
  - Os quadrados removidos podem ser cobertos por um triminó extra.
  - Então o tabuleiro original pode ser totalmente ladrilhado.

C.Q.D.

Indução: ideia dos ladrilhos



Conceitos de computação

Conceitos de computação

- Seção 1: Processamento
- Seção 2: Estruturas de dados básicas
  - Vetores
  - Listas
  - Pilhas e filas
  - Árvores
- Seção 3: Recursão

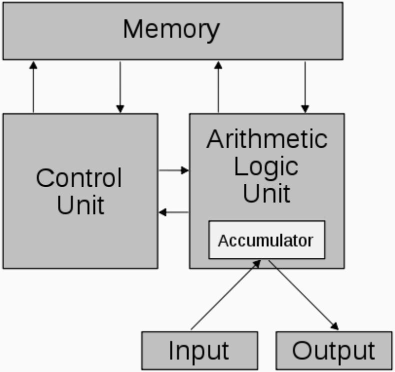
Materiais extras:

- Notas de aula da disciplina Programação Estruturada
- Notas de aula da disciplina Estruturas de Dados
- Notas de aula dessa disciplina

Processamento

Hardware e dispositivos

- **Hardware** são todos os dispositivos físicos que compõem um computador, como CPU, disco rígido, memória, etc.
- Esses dispositivos seguem uma organização básica, como na figura (Arq. de Von Neumann):



Unidade Central de Processamento - CPU

- É a CPU que executa os programas, **instrução** por vez.
- Uma instrução é um comando **atômico** que indica uma tarefa específica (soma, teste, desvio).
- Cada computador (processador) tem um conjunto particular de instruções.
    - Mas instruções básicas como leitura/escrita, soma, teste, desvio, são comuns a todos.

Unidade Central de Processamento - CPU

- Uma **palavra** é a parte mínima de dados que podem ser transferidos de/para a memória.
- Processadores de 32 bits têm palavras de 32 bits.
  - Assim, qualquer dado que será movimentado, somado, subtraído, deve ser descrito por 32 bits.
  - Obs.: $2^{32} = 4.294.967.296$.
- A frequência de um processador indica a velocidade com que ele consegue executar instruções.
  - De forma grosseira, um processador de 4.2GHz realiza 4.2G (4.2×10^9) instruções por segundo.

Resumindo

Computadores conseguem realizar instruções básicas sobre "dados pequenos" de forma extremamente rápida.

Estruturas de dados básicas

Estruturas de dados

Tipo abstrato de dados

Um **tipo abstrato de dados** é um conjunto de dados, as relações entre eles e as funções e operações que podem ser aplicadas aos dados.

Estrutura de dados

Uma **estrutura de dados** é uma implementação de um tipo abstrato de dados.

Vetores

Vetores

Um **vetor** é uma estrutura de dados linear que armazena uma sequência de elementos de mesmo tipo.

- Os elementos ocupam posições *contíguas* na memória, o que permite acesso direto a qualquer elemento por meio de um índice inteiro.
- Por isso, são todos referenciados por um mesmo identificador.

	1	2	3	4
A	12	99	37	24

- Problemas de interesse: inserir elemento, buscar elemento, remover elemento.

Listas

Listas

Uma **lista ligada** (ou **encadeada**) é uma estrutura de dados linear que armazena uma coleção de elementos.

- Cada elemento é armazenado em um **nó** ou **célula**.
- Cada nó guarda o endereço do próximo nó da lista.
- Têm-se acesso apenas ao primeiro nó: para acessar o k-ésimo, deve-se acessar o primeiro, que dá acesso ao segundo, e assim sucessivamente, até que o $(k-1)$-ésimo dá acesso ao k-ésimo.

- Problemas de interesse: inserir elemento, buscar elemento, remover elemento.

Pilhas

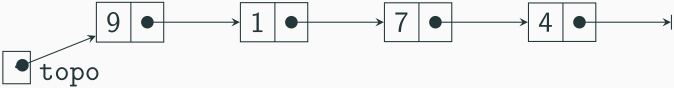
Pilha é um tipo abstrato de dados que oferece operações de adição e remoção de elementos.

- A remoção **deve ser** do elemento que está na coleção há menos tempo.
- Política de remoção "LIFO" (*last in, first out*).

Implementações mais comuns:

- Em vetor:

- Em lista:



Pilhas: exemplo

	1	2	3	4	5	6
P						
	topo					

	1	2	3	4	5	6
P	3					
		topo				

(EMPILHA(P, 3))

	1	2	3	4	5	6
P	3	5				
			topo			

(EMPILHA(P, 5))

	1	2	3	4	5	6
P	3	5	1			
				topo		

(EMPILHA(P, 1))

	1	2	3	4	5	6
P	3	5				
			topo			

(DESEMPILHA(P))

	1	2	3	4	5	6
P	3					
		topo				

(DESEMPILHA(P))

	1	2	3	4	5	6
P	3	8				
			topo			

(EMPILHA(P, 8))

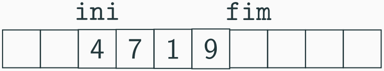
Filas

Fila é um tipo abstrato de dados que oferece operações de adição e remoção de elementos.

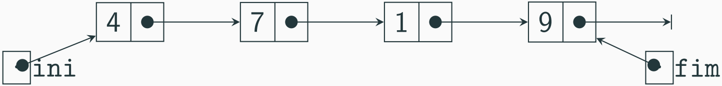
- A remoção **deve ser** do elemento que está na coleção há mais tempo.
- Política de remoção “FIFO” (*first in, first out*).

Implementações mais comuns:

- Em vetor:



- Em lista:



Filas: exemplo

	1	2	3	4	5	6
F						
	ini = fim					

	1	2	3	4	5	6	
F	3						(ENFILEIRA(F, 3))
	ini	fim					

	1	2	3	4	5	6	
F	3	5					(ENFILEIRA(F, 5))
	ini		fim				

	1	2	3	4	5	6	
F	3	5	1				(ENFILEIRA(F, 1))
	ini			fim			

	1	2	3	4	5	6	
F		5	1				(DESENFILEIRA(F))
		ini		fim			

	1	2	3	4	5	6	
F			1				(DESENFILEIRA(F))
			ini	fim			

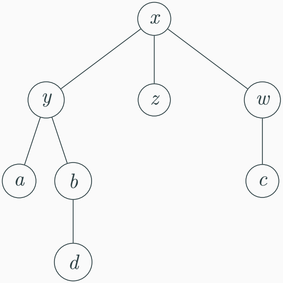
	1	2	3	4	5	6	
F			1	8			(ENFILEIRA(F, 8))
			ini		fim		

Árvores

Árvores são estruturas hierárquicas (não lineares) que armazenam um conjunto de elementos.

- Cada elemento é armazenado em um **nó**.
- Cada nó contém endereços para outros nós que estão abaixo dele na hierarquia, seus **filhos**.
- Temos acesso ao nó **raiz**, topo da hierarquia.

- Todo nó que não é raiz tem ligação com um único nó imediatamente acima dele na hierarquia, seu nó **pai**.
- Um nó que não tem filhos é chamado **folha**.

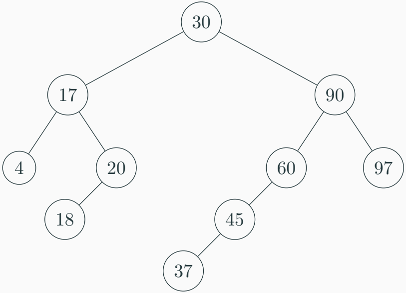


Árvores

- **Nível** ou **profundidade** de um nó x : quantidade de ligações no caminho entre x e a raiz.
- **Altura** de um nó x : quantidade de ligações no maior caminho entre x e uma folha.
- **Altura da árvore**: altura da raiz.

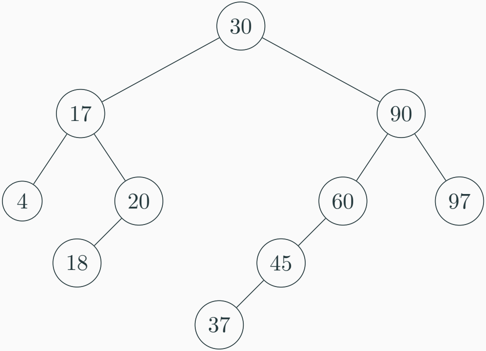
Árvores binárias

- Em uma **árvores binária** qualquer nó possui no máximo dois filhos: *filho esquerdo* e *filho direito*.
- Em uma **árvore binária de busca**, para cada nó x, todos os nós da subárvore esquerda de x possuem chaves menores que a de x e todos os nós da subárvore direita de x possuem chaves maiores que a de x.



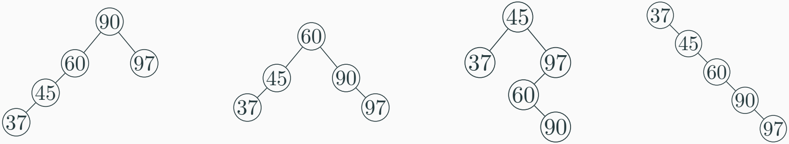
Árvores binárias de busca

- Operações de inserção, remoção e busca em árvores binárias de busca percorrem caminhos que vão da raiz até no máximo uma folha (seguindo por um número de ligações que é no máximo a altura).



Árvores binárias de busca

Um mesmo conjunto de elementos, dependendo da ordem na qual são inseridos em uma árvore, pode dar origem a árvores diferentes (com alturas diferentes):

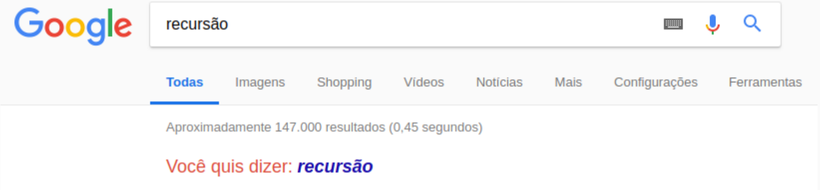


Árvores binárias de busca

- Se a árvore tem n nós e cada nível i tem no máximo 2^i nós, então
$$n \leq 2^0 + 2^1 + \cdots + 2^h$$
onde h é a altura da árvore.
- Como $2^0 + 2^1 + \cdots + 2^h = 2^{h+1} - 1$, então
$$2^{h+1} \geq n + 1\,,$$
de onde temos
$$h \geq \lceil \lg(n+1) \rceil$$
- Árvores binárias de busca **balanceadas** garantem altura bem próximo disso.
    - Árvores rubro-negras têm altura no máximo $2\lg(n+1)$.

Recursão

Recursão


Google
recursão
Todas
Imagens
Shopping
Vídeos
Notícias
Mais
Configurações
Ferramentas
Aproximadamente 147.000 resultados (0,45 segundos)
Você quis dizer: recursão

Recursão

Considere os seguintes problemas:

- Fatorial de um número n :

$$n! = n \times (n-1) \times (n-2) \times \cdots \times 2 \times 1$$

- A n-ésima potência de x :

$$x^n = \underbrace{x \times x \times x \times \cdots \times x}_{n\ fatores}.$$

- Máximo divisor comum (**MDC**) entre a e b :

$$MDC(a, b) = \text{maior inteiro positivo que divide } a \text{ e } b$$

Recursão

Todos eles podem ser definidos de forma **recursiva**, isto é, dependendo deles mesmos sobre entradas menores:

- **Fatorial** de um número n :

$$n! = \begin{cases} 1 & \text{se } n = 0 \\ n \times (n-1)! & \text{caso contrário} \end{cases}$$

- A n-ésima potência de x :

$$x^n = \begin{cases} 1 & \text{se } n = 0 \\ x \times x^{n-1} & \text{caso contrário} \end{cases}$$

- Máximo divisor comum (**MDC**) entre a e b :

$$MDC(a, b) = \begin{cases} a & \text{se } b = 0 \\ MDC(b, a \bmod b) & \text{caso contrário} \end{cases}$$

Recursão

- Fatorial de um número n:

  Função $\textsc{fat}(n)$

  Se $n == 0$ **então**

  Devolve 1

  Devolve $n \times \textsc{fat}(n-1)$

- A n-ésima potência de x:

  Função $\textsc{pot}(x,\ n)$

  Se $n == 0$ **então**

  Devolve 1

  Devolve $x \times \textsc{pot}(x,\ n-1)$

- Máximo divisor comum (**MDC**) entre a e b:

  Função $\textsc{mdc}(a,\ b)$

  Se $b == 0$ **então**

  Devolve a

  Devolve $\textsc{mdc}(b,\ a \% b)$

Recursão

- Um objeto é **recursivo** quando sua definição é parcialmente feita em termos dele mesmo.

Recursão

- Um objeto é **recursivo** quando sua definição é parcialmente feita em termos dele mesmo.
- Em programação, a recursividade é um mecanismo útil e poderoso que permite a uma função chamar a si mesma direta ou indiretamente.

Recursão

- Um objeto é **recursivo** quando sua definição é parcialmente feita em termos dele mesmo.
- Em programação, a recursividade é um mecanismo útil e poderoso que permite a uma função chamar a si mesma direta ou indiretamente.
- A ideia básica de um algoritmo recursivo consiste em diminuir sucessivamente o problema em um problema menor ou mais simples, até que o possamos resolver o problema reduzido de forma direta.

Recursão

- Um objeto é **recursivo** quando sua definição é parcialmente feita em termos dele mesmo.
- Em programação, a recursividade é um mecanismo útil e poderoso que permite a uma função chamar a si mesma direta ou indiretamente.
- A ideia básica de um algoritmo recursivo consiste em diminuir sucessivamente o problema em um problema menor ou mais simples, até que o possamos resolver o problema reduzido de forma direta.
    - Quando isso ocorre, atingimos uma **condição de parada**.

Recursão

- Um objeto é **recursivo** quando sua definição é parcialmente feita em termos dele mesmo.
- Em programação, a recursividade é um mecanismo útil e poderoso que permite a uma função chamar a si mesma direta ou indiretamente.
- A ideia básica de um algoritmo recursivo consiste em diminuir sucessivamente o problema em um problema menor ou mais simples, até que o possamos resolver o problema reduzido de forma direta.
  - Quando isso ocorre, atingimos uma **condição de parada**.
- Recursão e indução estão totalmente relacionadas!

Recursão

Descreva soluções recursivas para os seguintes problemas:

- Verificar se uma palavra é um palíndromo
- Buscar um elemento em um vetor
- Buscar o maior elemento de um vetor
- Somar os elementos de um vetor
- Ordenar um vetor
- Inverter uma palavra
- Encontrar soluções inteiras positivas para $x_1 + x_2 + x_3 + x_4 = C$
- Torre de Hanoi (3 pinos e n discos)