Teoria da Computação

Prof. Maicon R. Zatelli

Aula 1 - Introdução

Universidade Federal de Santa Catarina
Florianópolis - Brasil

2019/1

Introdução

Teoria da Complexidade

- Classifica os problemas em fáceis ou difíceis.

Teoria da Computabilidade

- Classifica os problemas em solucionáveis ou não-solucionáveis.

Teoria de Autômatos

- Definições e propriedades de modelos matemáticos de computação.

Noções Matemáticas e Terminologia

Conjuntos

Repetições não importam

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

Multiconjuntos

Repetições importam

- $\{1, 2, 3\} \neq \{1, 2, 2, 3\}$

Conjuntos Infinitos

Números naturais ($\mathbb{N}$)

- $\{1, 2, 3, ...\}$

Números inteiros ($\mathbb{Z}$)

- $\{..., -2, 1, 0, 1, 2, 3, ...\}$

Noções Matemáticas e Terminologia

Diagrama de Venn: $A \cap B$

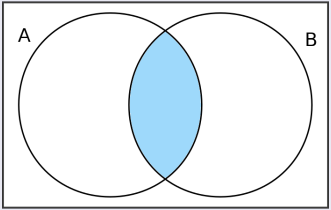
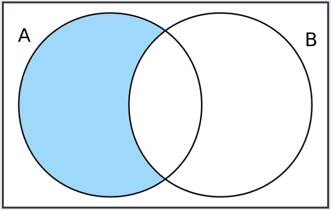


Diagrama de Venn: $A - B$



Noções Matemáticas e Terminologia

Sequência

Lista de objetos em alguma ordem, sendo que pode ser infinita e repetições não importam.

- $(7, 21, 100) = (7, 21, 21, 100)$

Uma **tupla** é uma sequência finita, sendo que uma 2-tupla é chamada de **par ordenado**.

Noções Matemáticas e Terminologia

Conjunto de Partes ou *Power Set*

É o conjunto de todos os subconjuntos de um conjunto (inclusive $\emptyset$)

- Seja A o conjunto $\{1, 2, 3\}$, então o conjunto de partes de A é $P(A) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$

Produto Cartesiano ou Produto Cruzado

O produto cartesiano de dois conjuntos, A e B, representado por $A \times B$, é o conjunto de todos os pares ordenados (a, b), tal que $a \in A$ e $b \in B$.

- Sejam os conjuntos $A = \{a, b\}$ e $B = \{1, 2\}$, então $A \times B = \{(a, 1), (a, 2), (b, 1), (b, 2)\}$

Noções Matemáticas e Terminologia

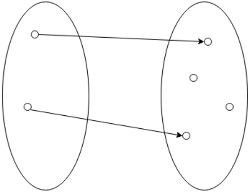
Função

É um objeto que produz uma relação de entrada-saída, onde, para uma mesma entrada sempre ocorre a mesma saída. Funções com um argumento são chamadas unárias; com dois argumentos são chamadas binárias; e assim sucessivamente.
Tipos de funções:

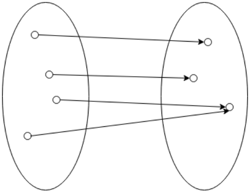
- Injetora (*one-to-one*, *injective*): todos os elementos do domínio (*domain*) possuem imagens (*range*) distintas.
- Sobrejetora (*onto*, *surjective*): não há elementos sobrando na imagem.
- Bijetora (*bijective*, *one-to-one and onto*, *correspondence*): todos os elementos do domínio possuem imagens distintas e não há elementos sobrando na imagem. Para ser bijetora, a função deve ser injetora e sobrejetora.

Noções Matemáticas e Terminologia

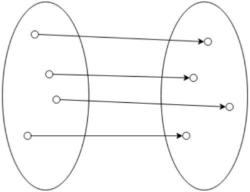
Injetora



Sobrejetora



Bijetora



Função unária

- $f(n) = n + 1$

n	f(n)
0	1
1	2
2	3
3	4

Função binária

- $f(i, j) = i + j$

f(i,j)	0	1	2	3
0	0	1	2	3
1	1	2	3	4
2	2	3	4	5
3	3	4	5	6

Noções Matemáticas e Terminologia

Predicado (ou Propriedade)

Um predicado ou propriedade é uma função cuja imagem é $\{True, False\}$.

- $par(4) = True$

Relação

Uma propriedade cujo domínio é um conjunto de k-tuplas $A \times A \times A \times ... \times A$ é chamada de relação.

- $R(a_1, a_2, ..., a_k) = True$

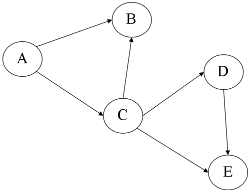
Propriedades de uma Relação

- Reflexiva: xRx
- Simétrica: $xRy \Rightarrow yRx$
- Transitiva: $xRy, yRz \Rightarrow xRz$

Se uma relação possui as três propriedades, a relação é chamada de relação de equivalência. Ex: relação *igual*

Grafos

Um **grafo** G é um par (V, E), onde V é um conjunto de **vértices** e E é um conjunto de **arestas**. Graficamente, um grafo pode ser representado por pontos e linhas, onde os pontos são os vértices e as linhas são as arestas. As arestas são pares ordenados (u, v), onde u é o vértice de origem e v é o vértice de destino.

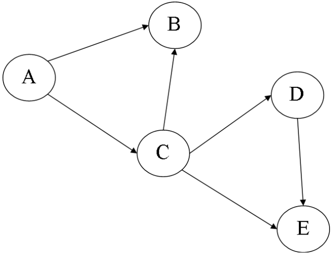


- $V = \{a, b, c, d, e\}$
- $E = \{(a, b), (a, c), (c, b), (c, d), (c, e), (d, e)\}$

Se as arestas do grafo **não** possuem uma ordem, dizemos que o grafo é **não-dirigido**, caso contrário ele será um grafo **dirigido** (ou digrafo).

Grafos

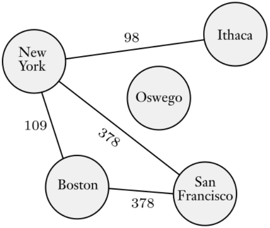
A quantidade de arestas que incidem em um vértice é o **grau** daquele vértice. Em um grafo dirigido, o **grau de entrada** refere-se a quantidade de arestas que chegam em determinado vértice e o **grau de saída** refere-se a quantidade de arestas que partem daquele vértice.



Se todos os vértices de um grafo possuem o mesmo grau, dizemos que o grafo é **regular**.

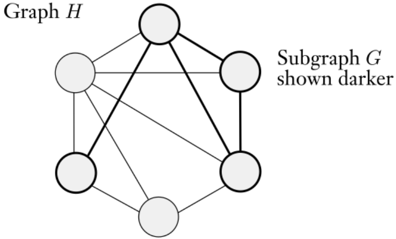
Grafos

- Quando a origem e o destino de uma aresta é um mesmo vértice, dizemos que ela é um **laço**.
- Quando mais de uma aresta conecta o mesmo par de vértices na mesma direção, dizemos elas são **paralelas**. Quando a direção for diferente, dizemos que elas são **anti-paralelas**.
- Um grafo é dito **simples** se ele não possui arestas paralelas e nem laços, caso contrário ele é chamado de **multigrafo**.
- Um grafo em que as arestas possuem valores associados é chamado de grafo **valorado** (ou grafo anotado).



Grafos

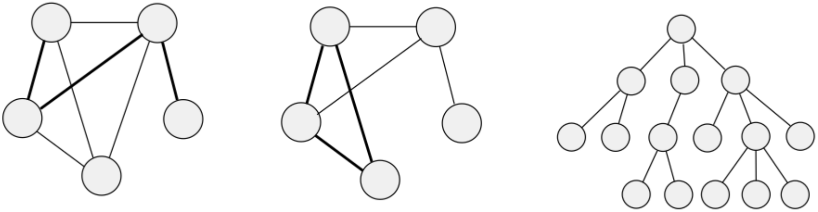
Um grafo G é um **subgrafo** de um grafo H, se ele possui um subconjunto de vértices de H e também as arestas de H correspondem às arestas de G para os vértices existentes em H e G.



Grafos

- Um **passeio** (ou *walk*) em um grafo é uma sequência de vértices em que se u e v são vértices na sequência, então (u, v) é uma aresta no grafo. No passeio, uma aresta pode estar presente várias vezes.
- Um **caminho** (ou *path*) é um passeio, mas sem arestas repetidas. Um caminho é dito simples se não há vértices repetidos.
- Um caminho é dito **fechado** (ou **ciclo**) se sua origem coincide com seu término. Um ciclo é simples se não há vértices repetidos nele, exceto pelo vértice de origem e destino.
- O **comprimento** de um caminho é o número de arestas nele contido.
- Um grafo é uma **árvore** se não há ciclos. Uma árvore pode conter um vértice (ou nó) **raiz**, sendo que os demais vértices (exceto pela raiz) que possuem grau 1 são chamadas de **folhas**.

Grafos



(a) (b) (c)

Grafos

Um grafo não-dirigido é dito **conectado** (ou **conexo**) se todos os vértices do grafo são alcançáveis a partir de qualquer vértice inicial, caso contrário o grafo é dito desconectado (ou desconexo). Os subconjuntos de vértices e arestas que formam um subgrafo conectado maximal do grafo original são chamados de **componentes conexas**.

Um grafo dirigido é dito **fortemente conectado** (ou **fortemente conexo**) se é possível alcançar todos os vértices do grafo a partir de qualquer vértice inicial (para todo vértice inicial). As **componentes fortemente conexas** de um grafo dirigido são os subgrafos fortemente conectados maximais do grafo dirigido.

Existem muitos outros conceitos e problemas envolvendo grafos:

- Detecção de pontes e nós de articulação
- Roteamentos e caminho mínimo
- Maximização de fluxo
- Árvores geradoras e árvores geradoras de custo mínimo
- Conectividade
- Planaridade, isomorfismo, coloração
- ...

Lógica Booleana

Lógica booleana é um sistema matemático construído em volta de dois valores: **verdadeiro** (V) e **falso** (F). Tais valores são chamados de valores booleanos. Pode-se manipular valores booleanos por meio de operações booleanas.

Operações Booleanas

- Negação (NOT): $\neg$
- Conjunção (AND): $\wedge$
- Disjunção (OR): $\vee$
- Ou exclusivo (XOR): $\otimes$
- Implicação (Se-Então): $\Rightarrow$
- Bi-implicação (equivalência) (Se-Somente-Se): $\Leftrightarrow$

Lógica Booleana

O modo como funcionam as operações booleanas pode ser visto facilmente por meio de tabelas-verdade:

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \otimes Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
V	V	F	V	V	F	V	V
V	F	F	F	V	V	F	F
F	V	V	F	V	V	V	F
F	F	V	F	F	F	V	V

Linguagens Formais

- **Símbolo**: menor elemento de uma linguagem. Ex: $2, a, \%, *$
- **Alfabeto** Σ (sigma): conjunto finito e não vazio de símbolos. Ex: $\Sigma = \{0, 1\}$
- **Cadeia** ou **Palavra**: sequência finita de símbolos sobre um alfabeto. Ex: $w = 1010101$
- **Palavra vazia** ε: palavra que não contém nenhum símbolo. É uma palavra sobre qualquer alfabeto)
- **Prefixo**: sequência de símbolos iniciais de w. Ex: $w = \textcolor{green}{0101}101$
- **Sufixo**: sequência de símbolos finais de w. Ex: $w = 010\textcolor{green}{1101}$
- **Subcadeia**: qualquer sequência contígua de w. Ex: $w = 010\textcolor{green}{110}1$

ε e w são prefixos, sufixos, e subcadeias de qualquer palavra w.

Linguagens Formais

Tamanho ou Comprimento $|w|$

É a quantidade de símbolos que compõe w.

- $|\varepsilon| = 0$
- $|abc| = 3$

Σ^k é o conjunto de todas as palavras de tamanho k sobre Σ.

- $\{0,1\}^2 = \{00, 01, 10, 11\}$
- $\{0,1\}^0 = \{\varepsilon\}$

Σ^* é o conjunto de todas as palavras sobre Σ, incluindo ε.

- $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 ...$

Σ^+ é o conjunto de todas as palavras sobre Σ, excluindo ε.

- $\Sigma^+ = \Sigma^1 \cup \Sigma^2 ...$

Reverso w^R

O reverso de uma palavra w, representado como w^R, é a palavra contendo a ordem inversa dos símbolos de w.

- $w = abc$, $w^R = cba$

Se $w = w^R$, dizemos que a palavra é um palíndromo.

Concatenação $u \circ v$

A concatenação das palavras u e v, representada por $u \circ v$, é a operação de adicionar os símbolos de v ao final de u, formando uma nova palavra w.

- Sejam as palavras $u = ab$ e $v = cd$, então $u \circ v = abcd$
- $v \circ \varepsilon = v$
- $\varepsilon \circ v = v$

w^k é a concatenação sucessiva de w por k vezes.

- Seja $w = ab$, então $w^3 = ababab$

Concatenação é uma operação associativa, ou seja:
$u \circ (v \circ t) = (u \circ v) \circ t$

Linguagens Formais

Linguagem Formal

É um conjunto de palavras sobre um alfabeto Σ, ou seja $L \subseteq \Sigma^*$

- $L =$ todas as palavras sobre $\Sigma = \{0, 1\}$ consistindo de k 0's seguidos por k 1's, ou seja: $L = \{\varepsilon, 01, 0011, 000111, ...\}$

A linguagem vazia $L = \emptyset$ é uma linguagem sobre qualquer alfabeto. Note que $\varepsilon \notin \emptyset$

A linguagem $L = \{\varepsilon\}$ consiste apenas na palavra vazia e é uma linguagem sobre qualquer alfabeto.

A linguagem $L = \Sigma^*$ consiste em todas as palavras sobre Σ e é uma linguagem sobre qualquer alfabeto.

Linguagens Formais

Linguagem Formal

Uma linguagem pode ser finita ou infinita.

- Se a linguagem for finita, basta enumerar todas as palavras. Ex: todas as palavras de tamanho três sobre um determinado alfabeto.
- Se a linguagem for infinita, deve-se utilizar representação de conjuntos da forma: $L = \{w|w \in \Sigma^*$e w possui determinada propriedade $P\}$
  - Ex: $L = \{w|w \in \{0,1\}*$ e $|w|$ é par $\}$

Linguagens Formais

Operações sobre Linguagens

As operações com linguagens formais são similares às operações da teoria dos conjuntos.

- **União**: $L_1 \cup L_2 = \{w | w \in L_1 \text{ ou } w \in L_2\}$
- **Intersecção**: $L_1 \cap L_2 = \{w | w \in L_1 \text{ e } w \in L_2\}$
- **Diferença**: $L_1 - L_2 = \{w | w \in L_1 \text{ e } w \notin L_2\}$
- **Concatenação**: $L_1 \circ L_2 = \{w | w = uv \text{ e } u \in L_1 \text{ e } v \in L_2\}$
    - Note que: $\emptyset \circ L = L \circ \emptyset = \emptyset$
    - Note que: $\{\varepsilon\} \circ L = L \circ \{\varepsilon\} = L$
- **Concatenação sucessiva** L^n: é a concatenação de L com ela própria n vezes.
- **Fecho de Kleene** L^*: são todas as palavras obtidas pela concatenação de zero ou mais palavras de L.
    - $L^* = \bigcup_{i=0}^{\infty} L^i = L^0 \cup L^1 \cup L^2 \cup \ldots$

Operações sobre Linguagens - Concatenação

Sejam as linguagens

- $L_1 = \{a, b, ab\}$
- $L_2 = \{0, 00, 10\}$

Então

- $L_1 \circ L_2 = \{a0, a00, a10, b0, b00, b10, ab0, ab00, ab10\}$

Operações sobre Linguagens - Concatenação Sucessiva

Seja a linguagem

- $L = \{0, 11, 10\}$

Então

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = \{0, 11, 10\} \circ \{\varepsilon\} = \{0, 11, 10\}$
- $L^2 = L \circ L^1 = \{0, 11, 10\} \circ \{0, 11, 10\} = \{00, 011, 010, 110, 1111, 1110, 100, 1011, 1010\}$

Operações sobre Linguagens - Fecho de Kleene

Seja a linguagem

- $L = \{aa, bb\}$

Então

- $L^0 = \{\varepsilon\}$
- $L^1 = \{aa, bb\}$
- $L^2 = \{aaaa, aabb, bbaa, bbbb\}$
- $L^3 = ...$

Assim

- $L^* = \{\varepsilon, aa, bb, aaaa, aabb, bbaa, bbbb\}$

Provas

- Definição: descreve objetos e noções que serão usados na prova. Ex: dois inteiros x e y são consecutivos se e somente se $y = x + 1$
- Afirmações ou Declarações: expressam propriedades que objetos possuem
- Prova: é um argumento lógico convincente de que uma declaração é verdadeira
- Teorema: é uma declaração que foi provada ser verdadeira
- Lemmas: são declarações provadas verdadeiras e que ajudam a provar outras declarações
- Colorários: são outras declarações que podemos concluir serem verdadeiras através de teoremas ou da prova destes teoremas

Provas

Formas de Sentenças a Serem Provadas

- Proposicional: $P(k)$, isto é, k possui a propriedade P.
  - Ex: $P(k)$ é verdade quando a soma de k e seu sucessor é ímpar
- Se-Então: $P \Rightarrow Q$, isto é, se P é verdade então Q é verdade.
  - Ex: se x e y são números inteiros consecutivos, então a soma de x e y é ímpar
- Se e somente Se: P *iff* Q, $P \Leftrightarrow Q$, isto é, deve-se provar $P \Rightarrow Q$ e $Q \Rightarrow P$.
  - Ex: seja x um número real, então $\lceil x \rceil = \lfloor x \rfloor$ se e somente se x é um inteiro
  - Neste caso, a declaração acima deve ser provada em dois passos:
    1. Assuma que $\lceil x \rceil = \lfloor x \rfloor$ e mostre que x é inteiro
    2. Assuma que x é inteiro e mostre que $\lceil x \rceil = \lfloor x \rfloor$

Provas

Métodos de Prova

- Prova por construção
- Prova direta
- Prova por contradição
- Prova por contraposição (ou prova indireta)
- Prova por indução

Prova por Construção

Na prova por construção deve-se provar uma declaração através da construção de um exemplo (objeto) concreto, mostrando que alguma coisa com certa propriedade existe. Ex: para mostrar que peixes azuis existem, basta mostrar um peixe azul.

Declaração

É possível computar x^n, com $n \geq 0$, com não mais de $2 * \log_2 n$ multiplicações.

Para provar que a declaração acima é verdade, podemos construir um algoritmo que compute x^n com até $2 * \log_2 n$ multiplicações.

Prova por Construção

Tal algoritmo pode ser este:

```
Potencia(x,n)

  if (n == 0) return 1

  if (n == 1) return x

  if (n % 2 == 0)
    k = Potencia(x, n/2)
    return k * k

  k = Potencia(x, (n-1)/2)
  return k * k * x
```

Prova Direta

A prova direta pode ser usada para provar sentenças da forma $P \Rightarrow Q$ e consiste em assumir a hipótese P como verdadeira e então mostrar que a conclusão Q é também verdadeira.

Declaração

A soma de dois números consecutivos é ímpar.

Prova Direta

A prova direta pode ser usada para provar sentenças da forma $P \Rightarrow Q$ e consiste em assumir a hipótese P como verdadeira e então mostrar que a conclusão Q é também verdadeira.

Declaração

A soma de dois números consecutivos é ímpar.

Precisamos de algumas definições:

- **Definição 1**: se x é um inteiro par, então $x = 2n$, para algum inteiro n
- **Definição 2**: se x é um inteiro ímpar, então $x = 2n + 1$, para algum inteiro n
- **Definição 3**: dois inteiros x e y são consecutivos se e somente se $y = x + 1$

Prova Direta

Reescrevendo a declaração na forma $P \Rightarrow Q$:

Declaração

Se x e y são inteiros consecutivos, então a soma de x e y é ímpar.

- $P =$ x e y são inteiros consecutivos
- $Q =$ a soma de x e y é ímpar

Prova Direta

Reescrevendo a declaração na forma $P \Rightarrow Q$:

Declaração

Se x e y são inteiros consecutivos, então a soma de x e y é ímpar.

- $P =$ x e y são inteiros consecutivos
- $Q =$ a soma de x e y é ímpar

Passos:

1. Assuma P verdadeira
2. Use P para mostrar que Q é verdadeira

Prova Direta

Prova:

Prova Direta

Prova:

- Assuma que x e y são números consecutivos

Prova Direta

Prova:

- Assuma que x e y são números consecutivos
- Assim, podemos escrever $y = x + 1$ (**Definição 3**)
  - **Definição 3**: dois inteiros x e y são consecutivos se e somente se $y = x + 1$

Prova Direta

Prova:

- Assuma que x e y são números consecutivos
- Assim, podemos escrever $y = x + 1$ (**Definição 3**)
  - **Definição 3**: dois inteiros x e y são consecutivos se e somente se $y = x + 1$
- Então, $x + y = x + x + 1$ (substituindo y por $x + 1$) e isso é igual a $2x + 1$

Prova Direta

Prova:

- Assuma que x e y são números consecutivos
- Assim, podemos escrever $y = x + 1$ (**Definição 3**)
  - **Definição 3**: dois inteiros x e y são consecutivos se e somente se $y = x + 1$
- Então, $x + y = x + x + 1$ (substituindo y por $x + 1$) e isso é igual a $2x + 1$
- Note agora que $2x + 1 = 2n + 1$, com $n = x$ e então, pela **Definição 2**, é um número ímpar, ou seja, $x + y$ é ímpar. $\square$
  - **Definição 2**: se x é um inteiro ímpar, então $x = 2n + 1$, para algum inteiro n

Prova por Contradição

A prova por contradição pode ser usada para provar sentenças da forma $P \Rightarrow Q$ e tenta-se mostrar que uma declaração é falsa e verdadeira ao mesmo tempo, assim gerando uma contradição.

Assume-se a hipótese P como verdadeira e a conclusão Q como falsa, ou seja, $P \Rightarrow \neg Q$. A contradição significa que alguma suposição é incorreta, ou P ou $\neg Q$. Por padrão, $\neg Q$ deveria ser falsa, uma vez que P é verdadeira.

Declaração

Se x e y são inteiros consecutivos, então a soma de x e y é ímpar.

- $P =$ x e y são inteiros consecutivos
- $Q =$ a soma de x e y é ímpar

Prova por Contradição

Passos:

1. Assuma P verdadeira
2. Assuma Q é falsa
3. Mostre uma contradição

Prova por Contradição

Prova:

Prova por Contradição

Prova:

- Assuma que x e y são números consecutivos (P)

Prova por Contradição

Prova:

- Assuma que x e y são números consecutivos (P)
- Assuma que $x + y$ não é ímpar ($\neg Q$)

Prova por Contradição

Prova:

- Assuma que x e y são números consecutivos (P)
- Assuma que $x + y$ não é ímpar ($\neg Q$)
- Se $x + y$ não é ímpar, então não há um inteiro n tal que $x + y = 2n + 1$.

Prova por Contradição

Prova:

- Assuma que x e y são números consecutivos (P)
- Assuma que $x + y$ não é ímpar ($\neg Q$)
- Se $x + y$ não é ímpar, então não há um inteiro n tal que $x + y = 2n + 1$.
- Porém, sabemos que $y = x + 1$ e podemos escrever $x + y = x + (x + 1) = 2x + 1$, ou seja, $n = x$

Prova por Contradição

Prova:

- Assuma que x e y são números consecutivos (P)
- Assuma que $x + y$ não é ímpar ($\neg Q$)
- Se $x + y$ não é ímpar, então não há um inteiro n tal que $x + y = 2n + 1$.
- Porém, sabemos que $y = x + 1$ e podemos escrever $x + y = x + (x + 1) = 2x + 1$, ou seja, $n = x$
- Ora, agora temos que $x + y \neq 2n + 1$ e também que $x + y = 2n + 1$, onde $n = x$

Prova por Contradição

Prova:

- Assuma que x e y são números consecutivos (P)
- Assuma que $x + y$ não é ímpar ($\neg Q$)
- Se $x + y$ não é ímpar, então não há um inteiro n tal que $x + y = 2n + 1$.
- Porém, sabemos que $y = x + 1$ e podemos escrever $x + y = x + (x + 1) = 2x + 1$, ou seja, $n = x$
- Ora, agora temos que $x + y \neq 2n + 1$ e também que $x + y = 2n + 1$, onde $n = x$
- Assim, $\neg Q$ é falso, "$x + y$ não é ímpar" é falso, ou seja, $x + y$ é ímpar. $\square$

Prova por Contraposição

A prova por contraposição pode ser usada para provar sentenças da forma $P \Rightarrow Q$ e tenta-se provar a contraposição de $P \Rightarrow Q$, que é $\neg Q \Rightarrow \neg P$.

A prova por contraposição se baseia na equivalência lógica de $P \Rightarrow Q$ e $\neg Q \Rightarrow \neg P$, ou seja, provando $\neg Q \Rightarrow \neg P$ estará também provando $P \Rightarrow Q$.

P	Q	$\neg P$	$\neg Q$	$P \Rightarrow Q$	$\neg Q \Rightarrow \neg P$
V	V	F	F	V	V
V	F	F	V	F	F
F	V	V	F	V	V
F	F	V	V	V	V

Este tipo de prova é útil quando a prova direta parece ser mais difícil.

Prova por Contraposição

Exemplo

- Se choveu (P), a calçada está molhada (Q).
- A calçada não está molhada ($\neg Q$)
- Então, não choveu ($\neg P$)

Prova por Contraposição

Passos:

1. Construa a contraposição da declaração
2. Assuma que $\neg Q$ é verdadeira
3. Use $\neg Q$ para mostrar que $\neg P$ é verdadeira

Declaração

Se x e y são inteiros consecutivos, então a soma de x e y é ímpar.

- $P =$ x e y são inteiros consecutivos
- $Q =$ a soma de x e y é ímpar

Prova por Contraposição

Prova:

Prova por Contraposição

Prova:

Declaração - Contraposição

Se a soma de x e y **não** é ímpar, então x e y **não** são inteiros consecutivos.

- $\neg P =$ x e y **não** são inteiros consecutivos
- $\neg Q =$ a soma de x e y **não** é ímpar

Prova por Contraposição

Prova:

Declaração - Contraposição

Se a soma de x e y **não** é ímpar, então x e y **não** são inteiros consecutivos.

- $\neg P = x$ e y **não** são inteiros consecutivos
- $\neg Q =$ a soma de x e y **não** é ímpar

- Assuma que a soma de x e y não é ímpar.

Prova por Contraposição

Prova:

Declaração - Contraposição

Se a soma de x e y **não** é ímpar, então x e y **não** são inteiros consecutivos.

- $\neg P$ = x e y **não** são inteiros consecutivos
- $\neg Q$ = a soma de x e y **não** é ímpar

- Assuma que a soma de x e y não é ímpar.
- Então, não há um inteiro n tal que $x + y = 2n + 1$ (**Definição 2**)

Prova por Contraposição

Prova:

Declaração - Contraposição

Se a soma de x e y **não** é ímpar, então x e y **não** são inteiros consecutivos.

- $\neg P$ = x e y **não** são inteiros consecutivos
- $\neg Q$ = a soma de x e y **não** é ímpar

- Assuma que a soma de x e y não é ímpar.
- Então, não há um inteiro n tal que $x + y = 2n + 1$ (**Definição 2**)
- Note que $2n + 1 = n + (n + 1)$ e portanto $n + 1$ é sucessor de n (**Definição 3**).

Prova por Contraposição

Prova:

Declaração - Contraposição

Se a soma de x e y **não** é ímpar, então x e y **não** são inteiros consecutivos.

- $\neg P =$ x e y **não** são inteiros consecutivos
- $\neg Q =$ a soma de x e y **não** é ímpar

- Assuma que a soma de x e y não é ímpar.
- Então, não há um inteiro n tal que $x + y = 2n + 1$ (**Definição 2**)
- Note que $2n + 1 = n + (n + 1)$ e portanto $n + 1$ é sucessor de n (**Definição 3**).
- Isso implica que x e y não podem ser consecutivos, ou seja, $x = n$ e $y = n + 1$. $\square$

Prova por Indução

Na prova por indução tenta-se mostrar que todos os elementos de um conjunto infinito possuem certa propriedade.

Ex 1: podemos mostrar que um programa funciona corretamente para todas as entradas.

Ex 2: seja $\mathbb{N}$, o conjunto dos números naturais $\{1, 2, 3, ...\}$ e seja P a propriedade que queremos provar, ou seja, queremos $P(k)$ (na forma proposicional) seja verdade para todo número natural k.

Prova por Indução

Partes da prova por indução:

Base

A partir de quando a indução funciona, por exemplo, $P(1)$, ou outro valor.

Passo da Indução

Prova-se para os demais casos a partir da base, isto é, para todo $P(i)$, com $i \geq 1$, se $P(i)$ é verdade para algum i, então isto implica que $P(i+1)$ também é verdade.

Pelo princípio da indução, provando-se a base da indução e o passo indutivo tem-se que $P(i)$ é verdadeiro para todo $i \geq base$.

Prova por Indução

A suposição de que $P(i)$ é verdadeira é chamada de **hipótese de indução**.

- Podemos ter uma hipótese de indução mais forte, como $P(j)$ é verdade para todo $j \leq i$.

Estrutura da prova por indução:

1. Prova-se a base da indução. Ex: provar que $P(1)$ é verdade
2. Prova-se o passo indutivo. Assuma que $P(i)$ é verdade para algum i e usa-se a suposição para mostrar que $P(i+1)$ é verdade.

Prova por Indução

Declaração

A soma de dois números consecutivos é ímpar.

Reescrevendo a declaração na forma $P(k)$:

Declaração

$P(k)$ é verdade quando a soma de k e seu sucessor é ímpar.

Prova por Indução

Prova:

Prova por Indução

Prova:

Base da indução: $P(1)$

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n + 1 = 3$, com $n = 1$, satisfazendo a **Definição 2**.

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n + 1 = 3$, com $n = 1$, satisfazendo a **Definição 2**.

Passo indutivo:

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n + 1 = 3$, com $n = 1$, satisfazendo a **Definição 2**.

Passo indutivo:

- Assuma que $P(k)$ é verdade para algum valor de k. (hipótese)

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n + 1 = 3$, com $n = 1$, satisfazendo a **Definição 2**.

Passo indutivo:

- Assuma que $P(k)$ é verdade para algum valor de k. (hipótese)
- Vamos mostrar que $P(k + 1)$ é verdade.

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n + 1 = 3$, com $n = 1$, satisfazendo a **Definição 2**.

Passo indutivo:

- Assuma que $P(k)$ é verdade para algum valor de k. (hipótese)
- Vamos mostrar que $P(k + 1)$ é verdade.
- Pela hipótese, sabemos que $k + (k + 1)$ é ímpar

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n+1=3$, com $n=1$, satisfazendo a **Definição 2**.

Passo indutivo:

- Assuma que $P(k)$ é verdade para algum valor de k. (hipótese)
- Vamos mostrar que $P(k+1)$ é verdade.
- Pela hipótese, sabemos que $k+(k+1)$ é ímpar
- Se somarmos 1 a k, seu sucessor será $(k+1)+1$ e assim temos que $(k+1)+(k+1+1)=P(k+1)$.

Prova por Indução

Prova:

Base da indução: $P(1)$

- A soma de 1 e 2 é 3 e 3 é ímpar, pois podemos encontrar um n onde $2n + 1 = 3$, com $n = 1$, satisfazendo a **Definição 2**.

Passo indutivo:

- Assuma que $P(k)$ é verdade para algum valor de k. (hipótese)
- Vamos mostrar que $P(k + 1)$ é verdade.
- Pela hipótese, sabemos que $k + (k + 1)$ é ímpar
- Se somarmos 1 a k, seu sucessor será $(k + 1) + 1$ e assim temos que $(k+1) + (k + 1+1) = P(k + 1)$.
- Além disso, note que $(k+1) + (k + 1+1) = P(k + 1) = 2(k + 1) + 1 = 2n + 1$, com $n = k + 1$ o que mostra que é um número ímpar (**Definição 2**). $\square$

Conclusão

Noções matemáticas e terminologias

- Conjuntos
- Funções
- Relações
- Grafos
- Lógica booleana
- Linguagens formais
- Métodos de prova

Referências

- Livro Sipser, Capítulo 0
- Livro Hopcroft, Capítulo 1