Indução Estrutural

Fundamentos de Algoritmos

INF05008

Indução Estrutural

- **Método de prova** usado em Lógica Matemática, **Ciência da Computação** e em diversos outros campos da Matemática

- Baseia-se nos **Axiomas de Peano**

  – **Axioma** é uma sentença ou proposição que **não é provada ou demonstrada**, mas é considerada como **óbvia**

    Ex.: *Axioma da igualdade*: em qualquer linguagem onde uma variável `x` é definida, `x = x`

Axiomas de Peano

- Conjunto de 9 axiomas para o conjunto dos números naturais $\mathbb{N}$

1. $\forall x \in \mathbb{N}$, $x = x$ (**Reflexividade**)
2. $\forall x, y \in \mathbb{N}$, se $x = y$, então $y = x$ (**Simetria**)
3. $\forall x, y, z \in \mathbb{N}$, se $x = y$ e $y = z$, então $x = z$ (**Transitividade**)
4. $\forall x, y$, se $x \in \mathbb{N}$ e $x = y$, então $y \in \mathbb{N}$
5. $0 \in \mathbb{N}$
6. Seja $S(x) = x + 1$ uma **função sucessor**, $\forall n \in \mathbb{N}, S(n) \in \mathbb{N}$
7. $\forall n \in \mathbb{N}$, $S(n) \neq 0$
8. $\forall m, n \in \mathbb{N}$, se $S(m) = S(n)$, então $m = n$ (**função sucessor é injetora**)

Axiomas de Peano

- Conjunto de 9 axiomas para o conjunto dos números naturais $\mathbb{N}$

1. $\forall x \in \mathbb{N}$, $x = x$ (**Reflexividade**)
2. $\forall x, y \in \mathbb{N}$, se $x = y$, então $y = x$ (**Simetria**)
3. $\forall x, y, z \in \mathbb{N}$, se $x = y$ e $y = z$, então $x = z$ (**Transitividade**)
4. $\forall a$ e b, se $a \in \mathbb{N}$ e $a = b$, então $b \in \mathbb{N}$
5. $0 \in \mathbb{N}$
6. Seja $S(x)$ uma **função sucessor**, $\forall n \in \mathbb{N}, S(n) \in \mathbb{N}$
7. $\forall n \in \mathbb{N}$, $S(n) \neq 0$
8. $\forall m, n \in \mathbb{N}$, se $S(m) = S(n)$, então $m = n$ (**função sucessor é injetora**)

Axiomas de Peano

- O último axioma é chamado de **axioma da indução**, e é usado para descrever todos os números naturais:

  9. Se K é um conjunto tal que

  – $0 \in K$, e

  – $\forall n \in \mathbb{N}$, se $n \in K$, então $S(n) \in K$

  então K contém todos os naturais (i.e., $\mathbb{N} \subseteq K$)

- Note que essa definição é **recursiva**: uma vez aplicada a um natural x, ela pode ser aplicada ao seu sucessor $S(x)$, depois ao sucessor de $S(x)$, e assim por diante, tornando possível **construir outros naturais a partir de naturais conhecidos**

Prova por Indução Matemática

- O axioma da indução é generalizado para representar **proposições** que são **válidas para todos os números naturais**:

  Se P é uma proposição tal que

  - **$P(0)$ é verdadeira (base), e**
  - **$\forall n \in \mathbb{N}$, se $P(n)$ é verdadeira, então $P(S(n))$ também é verdadeira (passo indutivo)**

  então $P(n)$ é verdadeira para todo $n \in \mathbb{N}$

- O uso da indução matemática nos permite provar que uma **proposição é verdadeira** para todos os naturais **sem que tenhamos de avaliar a proposição para cada elemento** do conjunto $\mathbb{N}$

Prova por Indução Matemática (cont.)

- Podemos, por exemplo, definir a operação de **adição entre naturais** como uma função $+ : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$, onde as seguintes proposições são verdadeiras para todo $a, b \in \mathbb{N}$:

  - $a + 0 = a$
    - $0 + 0 = 0 \Rightarrow 0 = 0$, e
    - Se $1 + 0 = 1 \Rightarrow 1 = 1$,
      então $S(1) + 0 = S(1) \Rightarrow S(1) = S(1)$
  - $a + S(b) = S(a + b)$
    - $0 + S(0) = S(0 + 0) \Rightarrow S(0) = S(0)$, e
    - Se $1 + S(2) = S(1 + 2) \Rightarrow 4 = S(3) \Rightarrow 4 = 4$,
      então $S(1)+S(2) = S(S(1)+2) \Rightarrow 2+3 = S(2+2) \Rightarrow 5 = S(4) \Rightarrow 5 = 5$

Prova por Indução Estrutural

- A indução estrutural utiliza o mesmo processo da indução matemática para provar que uma dada proposição $P(x)$ é válida para qualquer elemento de uma **estrutura de dados** definida de forma **recursiva**

- A prova de uma indução estrutural para $P(x)$ é uma prova de que:
  - $P(x)$ é válida para todas as *estruturas mínimas* (**base**), e que
  - Se $P(x)$ é verdadeira para todas as *subestruturas* de uma estrutura S, então $P(x)$ também é verdadeira para S (**passo indutivo**)

Exemplo: Listas

- Uma **lista** L é uma estrutura de dados que pode ter um tamanho $tam(L)$ de valor **arbitrário**
- Uma lista **começa** sempre **vazia** ($[]$) até que um **primeiro elemento (**p**)** seja inserido (i.e., lista $[]$ é a **estrutura mínima**)
- **Novos elementos** são adicionados sempre **após o fim da lista**
- Uso de listas introduz uma **relação de ordem parcial** $\leq: Lista \times Lista$ na qual, dadas duas listas L e M, $L \leq M$ é verdadeira se L é a sublista final de M (**sublista** f)

Indução Estrutural em Listas

- Assim como a indução matemática permite construir novos naturais a partir de naturais conhecidos, a indução estrutural nos permite **construir novas listas a partir de uma lista existente** $\Rightarrow$ **operação de concatenação ++**

- Prova de uma proposição $P(l)$ consiste em demonstrar que:

  – $P([])$ é verdadeira (**base**), e

  – Dadas duas listas L e M, se $P(L)$ é verdadeira e $L \leq M$, então $P(M)$ também é verdadeira (**passo indutivo**)

Indução Estrutural em Listas (cont.)

- Propriedades de listas
  - $tam([]) = 0$ [TAM1]
  - $tam(p : f) = 1 + tam(f)$ [TAM2]
  - $[]$ ++ $lista = lista$ [CONC1]
  - $(p : f)$ ++ $lista = (p : (f$ ++ $lista))$ [CONC2]

Indução Estrutural em Listas (cont.)

- *Provar que a proposição* $P(l) : tam(l \text{ ++ } M) = tam(l) + tam(M)$ *é verdadeira para qualquer lista* l

- **Base**: demonstrar que $P(l)$ é verdadeira quando l é vazia

1. $tam(l \text{ ++ } M) = tam(l) + tam(M)$
2. $tam([] \text{ ++ } M) = tam([]) + tam(M)$
3. $tam(M) = tam([]) + tam(M)$ [usando CONC1]
4. $tam(M) = 0 + tam(M)$ [usando TAM1]
5. $tam(M) = tam(M)$

Indução Estrutural em Listas (cont.)

- **Passo indutivo**: demonstrar que $P(l)$ é verdadeira para uma lista l não vazia

  1. $tam(l \mathbin{++} M) = tam(l) + tam(M)$
  2. $tam((p : f) \mathbin{++} M) = tam(p : f) + tam(M)$
  3. $tam(p : (f \mathbin{++} M)) = tam(p : f) + tam(M)$ [usando CONC2]
  4. $1 + tam(f \mathbin{++} M) = tam(p : f) + tam(M)$ [usando TAM2]
  5. $1 + tam(f \mathbin{++} M) = 1 + tam(f) + tam(M)$ [usando TAM2]
  6. $tam(f \mathbin{++} M) = tam(f) + tam(M)$