Indução e Recursão: Indução Matemática Fraca e Forte, Indução Estrutural, Algoritmos Recursivos

Área de Teoria DCC/UFMG

PAA: Revisão de Conceitos de Lógica Computacional

2022/02

Introdução

Indução e recursão: Introdução

- Indução e recursão são técnicas essenciais da Matemática Discreta e têm inúmeras aplicações em Ciência da Computação.
- Muitas afirmações matemáticas estabelecem que uma certa propriedade é satisfeita por todo inteiro positivo n:

  1. $n! \leq n^n$
  2. $n^3 - n$ é divisível por 3.
  3. se um conjunto tem n elementos, seu conjunto potência tem 2^n elementos.

Aqui vamos ver uma técnica poderosa para demonstrar este tipo de resultado: a <u>indução matemática</u>.

Indução Matemática (Fraca)

Princípio da indução matemática: Intuição

- Imagine que você esteja diante de uma escada de infinitos degraus, e você se pergunta: *"Será que eu consigo alcançar qualquer degrau dessa escada?"*
- Você sabe que
  1. você consegue alcançar o primeiro degrau, e
  2. se você alcançar um degrau qualquer, você consegue alcançar o próximo degrau.
- Usando as regras acima, você pode deduzir que:
  1. você consegue alcançar o primeiro degrau: pela regra 1;
  2. você consegue alcançar o segundo degrau: pela regra 1, depois regra 2;
  3. você consegue alcançar o terceiro degrau: regra 1, depois regra 2 por duas vezes;
  4. ...
  5. você consegue alcançar o n-ésimo degrau: regra 1, depois regra 2 por $n-1$ vezes.
- Logo, você pode concluir que pode alcançar todos os degraus da escada!

Princípio da indução matemática (fraca)

- Para mostrar que uma propriedade $P(n)$ vale para todos os inteiros positivos n, uma **demonstração** que utilize o **princípio da indução matemática (fraca)** possui duas partes:

  Demonstração por indução fraca:

  Passo base: Demonstra-se $P(1)$.

  Passo indutivo: Demonstra-se que, para qualquer inteiro positivo k, se $P(k)$ é verdadeiro, então $P(k+1)$ é verdadeiro.

- A premissa do passo indutivo ($P(k)$ é verdadeiro) é chamada de **hipótese de indução** ou **I.H.**

- O princípio da indução matemática pode ser expresso como uma regra de inferência sobre os números inteiros:

$$\left(\underbrace{P(1)}_{\text{Passo base}} \quad \land \quad \underbrace{\forall k \in \mathbb{Z}^+ : (P(k) \to P(k+1))}_{\text{Passo indutivo}} \right) \quad \to \quad \underbrace{\forall n \in \mathbb{Z}^+ : P(n)}_{\text{Conclusão}}$$

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 1}}$ Se n é um inteiro positivo, então $1 + 2 + \cdots + n = n(n+1)/2$.

  Demonstração. Seja $P(n)$ a proposição *"a soma dos n primeiros inteiros positivos é* $n(n+1)/2$*"*.

  Passo base: $P(1)$ é verdadeiro porque

  $$1 = \frac{1(1+1)}{2}.$$

  Passo indutivo: Assuma que $P(k)$ seja verdadeiro para um inteiro positivo arbitrário k. Ou seja, a nossa hipótese de indução é de que, para um inteiro positivo k arbitrário:

  $$1 + 2 + \cdots + k = \frac{k(k+1)}{2}.$$

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 1}}$ (Continuação)

Sob a hipótese de indução, deve-se mostrar que $P(k+1)$ é válido, ou seja:

$$1 + 2 + \cdots + k + (k+1) = \frac{(k+1)[(k+1)+1]}{2} = \frac{(k+1)(k+2)}{2}.$$

Podemos, então, derivar

$$\begin{aligned} 1 + 2 + \cdots + k + (k+1) &= \frac{k(k+1)}{2} + (k+1) && \text{(pela I.H.)} \\ &= \frac{k(k+1) + 2(k+1)}{2} \\ &= \frac{(k+1)(k+2)}{2}, \end{aligned}$$

de onde concluímos o passo indutivo.

Como concluímos com sucesso tanto o passo base quanto o passo indutivo, mostramos por indução que $\forall n \in \mathbb{Z}^+ : P(n)$, ou seja, que $1 + 2 + \cdots + n = n(n+1)/2$ para todo inteiro positivo n. $\square$

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 2}}$ Desenvolva uma conjectura de uma fórmula equivalente à soma dos n primeiros inteiros ímpares.

  Então, demonstre sua conjectura usando indução matemática.

  Solução.

  Vamos começar testando alguns exemplos com valores de n:

$$
\begin{aligned}
n = 1: &\quad 1 \\
n = 2: &\quad 1 + 3 = 4 \\
n = 3: &\quad 1 + 3 + 5 = 9 \\
n = 4: &\quad 1 + 3 + 5 + 7 = 16 \\
n = 5: &\quad 1 + 3 + 5 + 7 + 9 = 25 \\
\ldots: &\quad 1 + 3 + 5 + 7 + 9 + 11 + 13 + ... = ?
\end{aligned}
$$

  Qual padrão podemos tentar inferir a partir dos exemplos acima?

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 2}}$ (Continuação)

Assim, chegamos a uma conjectura razoável, que tentaremos demonstrar:

"A soma dos n primeiros inteiros positivos ímpares é n^2."

Demonstração. Seja $P(n)$ a proposição *"A soma dos n primeiros inteiros positivos ímpares é n^2"*.

Passo base: $P(1)$ é verdadeiro porque o primeiro inteiro positivo ímpar é 1, o que é igual a 1^2.

Passo indutivo: Assuma que $P(k)$ seja verdadeiro para um inteiro positivo arbitrário k.

Note que o k-ésimo inteiro positivo ímpar é dado por $2k - 1$.

Logo, a hipótese de indução é:

$$1 + 3 + 5 + \cdots + (2k - 1) = k^2.$$

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 2}}$ (Continuação)

  Queremos mostrar que $\forall k \in \mathbb{Z}^+ : (P(k) \rightarrow P(k+1))$, onde $P(k+1)$ é:

  $$1 + 3 + \cdots + (2k-1) + (2(k+1)-1) = (k+1)^2.$$

  Logo, podemos derivar

  $$\begin{aligned} 1 + 3 + \cdots + (2k-1) + (2(k+1)-1) &= k^2 + (2(k+1)-1) \quad \text{(pela I.H.)} \\ &= k^2 + 2k + 1 \\ &= (k+1)^2, \end{aligned}$$

  de onde concluímos o passo indutivo.

  Como concluímos com sucesso o passo base e o passo indutivo, mostramos por indução que $\forall n \in \mathbb{Z}^+ : P(n)$, ou seja, que a soma dos n primeiros ímpares positivos é n^2. $\square$

Exemplos de uso de indução matemática (fraca)

- No caso geral, podemos usar a indução para mostrar que uma propriedade vale para qualquer subconjunto $\{n \in \mathbb{Z} \mid n \geq n_0\}$ dos inteiros, em que $n_0 \in \mathbb{Z}$.

- $\boxed{\text{Exemplo 3}}$ Para todo inteiro $n \geq 4$, $2^n < n!$.

Demonstração. Seja $P(n)$ a proposição "$2^n < n!$".

Passo base: $P(4)$ é verdadeiro porque $2^4 = 16$ é menor que $4! = 24$.

Passo indutivo: Assuma que $P(k)$ seja verdadeiro para um inteiro positivo arbitrário $k \geq 4$, ou seja, a hipótese de indução é que, para um inteiro arbitrário $k \geq 4$,

$$2^k < k!.$$

Sob esta hipótese, queremos mostrar $P(k+1)$, ou seja,

$$2^{k+1} < (k+1)!$$

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 3}}$ (Continuação)

  Para isto, podemos derivar

$$
\begin{aligned}
2^{k+1} &= 2(2^k) \\
&< 2(k!) && \text{(pela I.H.)} \\
&< (k+1)k! && \text{(já que } k \geq 4\text{)} \\
&= (k+1)!,
\end{aligned}
$$

  de onde concluímos o passo indutivo.

  Logo, por indução mostramos que $\forall n \in \mathbb{Z}, n \geq 4 : P(n)$, ou seja, que $2^n < n!$ para todo inteiro $n \geq 4$. $\square$

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 4}}$ Para todo inteiro não-negativo n, se um conjunto possui n elementos, então este conjunto possui 2^n subconjuntos.

  Demonstração. Seja $P(n)$ a proposição *"todo conjunto de n elementos possui 2^n subconjuntos"*.

  Passo base: $P(0)$ é verdadeiro porque o único conjunto de 0 elementos é o conjunto vazio $\emptyset$, que possui somente $2^0 = 1$ subconjunto (ele mesmo).

  Passo indutivo: Assuma que $P(k)$ seja verdadeiro para um inteiro não-negativo arbitrário k, ou seja, a hipótese de indução é:

  "Todo conjunto de k elementos possui 2^k subconjuntos."

  Sob a I.H., queremos demonstrar $P(k+1)$, ou seja, que

  "Todo conjunto de $k+1$ elementos possui 2^{k+1} subconjuntos."

Exemplos de uso de indução matemática (fraca)

- $\boxed{\text{Exemplo 4}}$ (Continuação)

Para mostrar isto, seja T um conjunto qualquer de $k+1$ elementos. Então é possível escrever T como $S \cup \{a\}$, onde

- a é um elemento qualquer de T;
- $S = T - \{a\}$ e, portanto, $|S| = k$.

Note que os subconjuntos de T podem ser obtidos da seguinte forma.

Para cada subconjunto X de S, existem exatamente dois suconjuntos de T: o subconjunto X (em que a não aparece) e o subconjunto $X \cup \{a\}$ (em que a aparece). Logo o número de subconjuntos de T é o dobro do número de subconjuntos de S. Pela hipótese indutiva, S tem 2^k subconjuntos, logo T possui $2 \cdot 2^k = 2^{k+1}$ subconjuntos. Isto conclui o passo indutivo.

Logo, por indução mostramos que $\forall n \in \mathbb{N} : P(n)$, ou seja, que todo conjunto de n elementos possui 2^n subconjuntos. $\square$

Quando usar indução matemática

- O princípio da indução pode ser utilizado para demonstrar propriedades dos números inteiros (se elas forem verdadeiras).
- O princípio da indução não pode ser utilizado para descobrir propriedades dos números inteiros.

  A propriedade geralmente é descoberta usando um outro método (talvez até tentativa e erro).

  Uma vez que uma propriedade tenha sido conjecturada, a indução pode ser usada para demonstrá-la (caso a propriedade seja mesmo verdadeira).

Modelo de demonstração por indução matemática (fraca)

1. Expresse a afirmação a ser demonstrada na forma *"para todo inteiro $n \geq b$, $P(n)$"*, onde b é um inteiro fixo.
2. Escreva "Passo base." e mostre que $P(b)$ é verdadeiro, se certificando de que o valor correto de b foi utilizado. Isto conclui o passo base.
3. Escreva as palavras "Passo indutivo."
4. Escreva claramente a hipótese indutiva, na forma "Assuma que $P(k)$ seja verdadeiro para um inteiro arbitrário fixo $k \geq b$."
5. Escreva o que precisa ser demonstrado sob a suposição de que a hipótese de indução é verdadeira. Ou seja, escreva o que $P(k+1)$ significa.
6. Demonstre a afirmação $P(k+1)$ utilizando o fato de que $P(k)$ é verdadeiro. Certifique-se de que sua demonstração é válida para qualquer $k \geq b$.
7. Identifique claramente as conclusões do passo indutivo, e conclua-o escrevendo, por exemplo, "isto completa o passo de indução".
8. Completados o passo base e o passo indutivo, escreva a conclusão da demonstração: que, por indução matemática, $P(n)$ é verdadeiro para todos os inteiros $n \geq b$.

Princípio da indução matemática (fraca): Erros comuns

- Como em qualquer outra técnica de demonstração, o princípio da indução matemática deve ser usado com cautela para evitar erros.

- Em particular, para que a demonstração por indução esteja correta é preciso demonstrar ambos o passo base e o passo indutivo.

  Se um dos dois passos não for demonstrado, o resultado não está garantido!

- $\boxed{\text{Exemplo 5}}$ Imagine que tenhamos a conjectura de que o predicado $P(n)$ definido como *"10^n é múltiplo de 7"* é verdadeiro para todo $n \in \mathbb{N}$.

  Se quisermos demonstrar esta afirmação por indução:

  a) É possível demonstrar o passo indutivo?

  b) É possível demonstrar o passo base?

  c) A demonstração por indução pode ser concluída com sucesso?

Princípio da indução matemática (fraca): Erros comuns

- $\boxed{\text{Exemplo 5}}$ (Continuação)

Solução.

a) Vamos começar pelo passo indutivo.

Passo indutivo: Assuma como hipótese indutiva que $P(k)$ seja verdadeiro para um inteiro $k \geq 0$ arbitrário, ou seja, que 10^k é divisível por 7. Sob a I.H., queremos mostrar que $P(k+1)$ também deve ser verdadeiro, ou seja, que 10^{k+1} é divisível por 7.

Se 10^k é divisível por 7, então existe um inteiro r tal que que $10^k = 7r$.

Logo podemos derivar

$$\begin{aligned} 10^{k+1} &= 10 \cdot 10^k \\ &= 10 \cdot 7r && \text{(pela I.H.)} \\ &= 7(10r) \end{aligned}$$

e, portanto, 10^{k+1} é divisível por 7, o que conclui o passo indutivo com sucesso.

Princípio da indução matemática (fraca): Erros comuns

- $\boxed{\text{Exemplo 5}}$ (Continuação)

  b) Agora olharemos o passo base.

  Passo base: Queremos mostrar $P(0)$, ou seja, que $10^0 = 1$ é divisível por 7. Mas isso é claramente falso.

  Logo o passo base não é válido.

  c) Por fim concluímos que a demonstração por indução não foi completada com sucesso, pois, apesar de o passo indutivo ter sido demonstrado, o passo base não foi.

  (Na verdade, o predicado $P(n)$ é falso para todo $n \in \mathbb{N}$!)

  $\lhd$

Indução Matemática (Forte)

Princípio da indução matemática (forte): Introdução

- O princípio de indução que vimos até agora é conhecido como o **princípio da indução matemática fraca**:

$$\left(\underbrace{P(1)}_{\text{Passo base}} \quad \wedge \quad \underbrace{\forall k \in \mathbb{Z}^+ : (\overbrace{P(k)}^{\text{I.H.}} \rightarrow P(k+1))}_{\text{Passo indutivo}} \right) \quad \rightarrow \quad \underbrace{\forall n \in \mathbb{Z}^+ : P(n)}_{\text{Conclusão}}$$

  Ele recebe este nome de indução "fraca" porque a hipótese de indução (I.H.) do passo indutivo é apenas que $P(k)$ seja verdadeiro para algum k.

- Às vezes é complicado usar a indução fraca para demonstrar um resultado, e podemos recorrer ao **princípio da indução matemática forte**.
  - Neste princípio, a hipótese de indução do passo indutivo é de que $P(j)$ é válido para todo $1 \leq j \leq k$.

Princípio da indução matemática (forte)

- Para mostrar que uma propriedade $P(n)$ vale para todos os inteiros positivos n, uma **demonstração** que utilize **princípio da indução matemática (forte)** possui duas partes:

  Demonstração por indução forte:

  Passo base: Demonstra-se $P(1)$;

  Passo indutivo: Demonstra-se que, para qualquer inteiro positivo k, se $P(j)$ é verdadeiro para todo $1 \leq j \leq k$, então $P(k+1)$ é verdadeiro.

- A **hipótese de indução** ou **I.H.** da indução forte é $P(1) \land P(2) \land \ldots \land P(k)$ são todos verdadeiros.

- O princípio da indução matemática forte pode ser expresso como uma regra de inferência sobre os números inteiros:

$$\left(\underbrace{P(1)}_{\text{Passo base}} \land \underbrace{\forall k \in \mathbb{Z}^+ : (P(1) \land P(2) \land \ldots \land P(k) \rightarrow P(k+1))}_{\text{Passo indutivo}} \right) \rightarrow \underbrace{\forall n \in \mathbb{Z}^+ : P(n)}_{\text{Conclusão}}$$

Princípio da indução matemática forte: Intuição

- Imagine que você esteja diante de uma escada de infinitos degraus, e você novamente se pergunta: *"Será que eu consigo alcançar qualquer degrau dessa escada?"*

- Mas, desta vez, você sabe que:

  1. você consegue alcançar o primeiro degrau e também o segundo degrau, e
  2. se você alcançar um degrau qualquer, você consegue alcançar dois degraus acima (ou seja, você pode subir degraus de dois em dois).

- Você consegue usar a indução fraca para verificar que conseguimos alcançar qualquer degrau dessa escada?

Princípio da indução matemática forte: Intuição

- Vamos tentar responder à pergunta usando indução forte.

  Vamos chamar de $P(n)$ a proposição *"Eu consigo alcançar o n-ésimo degrau da escada"*.

  Passo base: $P(1)$ é verdadeiro porque eu consigo alcançar o primeiro degrau. O mesmo vale para $P(2)$.

  Passo indutivo: Assumamos como hipótese de indução que para um $k \geq 2$, as proposições $P(1), P(2), \ldots, P(k)$ são todas verdadeiras. Queremos mostrar que $P(k+1)$ também é verdadeiro, ou seja, que podemos alcançar também o $(k+1)$-ésimo degrau.

  Para ver que podemos alcançar o degrau $k+1$, note que pela I.H. alcançamos todos os degraus entre 1 e k (para $k \geq 2$), e, em particular, o degrau $k-1$. Como alcançamos $k-1$ e a regra 2 diz que uma vez que tenhamos alcançado um degrau podemos alcançar dois degraus acima, podemos alcançar o degrau $k+1$. E assim termina o passo indutivo.

  Dessa forma, a demonstração por indução forte está completa.

Exemplos de uso de indução matemática (forte)

- $\boxed{\text{Exemplo 6}}$ Toda postagem de 12 centavos ou mais pode ser feita usando apenas selos de 4 centavos e selos de 5 centavos.

  Demonstração. Seja $P(n)$ a proposição *"qualquer postagem de n centavos pode ser feita usando apenas selos de* 4 *centavos e selos de* 5 *centavos"*.

  Passo base: Vamos precisar de quatro casos base:

  - $P(12)$ é verdadeiro porque podemos usar três selos de 4 centavos;
  - $P(13)$ é verdadeiro porque podemos usar dois selos de 4 centavos e um selo de 5 centavos;
  - $P(14)$ é verdadeiro porque podemos usar um selos de 4 centavos e dois selos de 5 centavos; e
  - $P(15)$ é verdadeiro porque podemos usar 3 selos de 5 centavos;

  Isto completa o passo base.

Exemplos de uso de indução matemática (forte)

- $\boxed{\text{Exemplo 6}}$ (Continuação)

Passo indutivo: A hipótese de indução é que $P(j)$ é verdadeiro para $12 \leq j \leq k$, onde k é um inteiro $k \geq 15$. Ou seja, a I.H. é que toda postagem de valores entre 12 centavos e k centavos pode ser feita usando selos de 4 e 5 centavos apenas.

Para completar o passo indutivo, vamos mostrar que, sob a I.H., $P(k+1)$ é verdadeiro, ou seja, que uma postagem de $k+1$ centavos pode ser feita usando-se apenas selos de 4 e 5 centavos.

Pela I.H., $P(k-3)$ é verdadeiro porque $k-3 \geq 12$ e para todo $12 \leq j \leq k$ temos $P(j)$ verdadeiro. Logo, existe uma maneira de postar $k-3$ centavos usando apenas selos de 4 e 5 centavos. Para postar $k+1$ centavos, basta acrescentar à postagem possível para $k-3$ centavos um selo de 4 centavos.

Isto concluímos o passo indutivo e a demonstração. $\square$

Definições Recursivas e Indução Estrutural

Recursão: Introdução

- Algumas vezes não é fácil definir um objeto explicitamente, mas é relativamente mais fácil definí-lo em termos de si próprio.

  Por exemplo:

  1. Definição dos números naturais em termos de números naturais:
     - 0 é um número natural;
     - o sucessor de um número natural é um número natural.

- A definição de um objeto em termos de si próprio é chamada **definição recursiva**.

- A **recursão** é muito utilizada para definir, por exemplo:

  1. funções,
  2. sequências,
  3. conjuntos, e
  4. algoritmos.

Definição recursiva de funções

- Uma **definição recursiva de uma função** com domínio nos números inteiros não-negativos tem duas partes:

  Definição recursiva de função:

  Passo base: Especifica-se o valor da função em 0.

  Passo recursivo: Especifica-se uma regra para encontrar o valor da função em um inteiro qualquer baseada no valor da função em inteiros menores.

- Lembre-se de que uma função $f(n)$ dos inteiros não-negativos para os reais é equivalente a uma sequência

$$a_0, \quad a_1, \quad a_2, \quad \ldots,$$

  onde a_i é um número real para cada inteiro não-negativo i.

  Logo, definir uma sequência $a_0, a_1, a_2, \ldots$ de números reais de forma recursiva é equivalente a definir uma função recursiva dos inteiros não-negativos para os reais.

Definição recursiva de funções

- $\boxed{\text{Exemplo 7}}$ Encontre uma definição recursiva para a função fatorial $f(n) = n!$, e compute $f(5)$ usando sua definição.

  Solução. Uma definição recursiva para $f(n) = n!$ é:

$$\begin{cases} f(0) = 1, \\ f(n) = n \cdot f(n-1), \quad n \geq 1 \end{cases}$$

  Podemos então calcular $f(5)$ como:

$$\begin{aligned} f(5) &= 5 \cdot f(4) \\ &= 5 \cdot (4 \cdot f(3)) \\ &= 5 \cdot (4 \cdot (3 \cdot f(2))) \\ &= 5 \cdot (4 \cdot (3 \cdot (2 \cdot f(1)))) \\ &= 5 \cdot (4 \cdot (3 \cdot (2 \cdot (1 \cdot f(0))))) \\ &= 5 \cdot (4 \cdot (3 \cdot (2 \cdot (1 \cdot 1)))) \\ &= 120. \end{aligned}$$

◁

Definição recursiva de funções

- $\boxed{\text{Exemplo 8}}$ Seja a um real não-nulo qualquer. Encontre uma definição recursiva para a função $f(n) = a^n$, tendo como domínio os naturais, e compute $f(3)$ usando sua definição.

  Solução. Uma definição recursiva para $f(n) = a^n$ é:

  $$\begin{cases} f(0) = 1, \\ f(n) = a \cdot f(n-1), \quad n \geq 1 \end{cases}$$

  Podemos então calcular $f(3)$ como:

  $$\begin{aligned} f(3) &= a \cdot f(2) \\ &= a \cdot (a \cdot f(1)) \\ &= a \cdot (a \cdot (a \cdot f(0))) \\ &= a \cdot (a \cdot (a \cdot 1)) \\ &= a^3. \end{aligned}$$

$\triangleleft$

Definição recursiva de funções

- $\boxed{\text{Exemplo 9}}$ Outros exemplos de definições recursivas:

Somatório:	$\begin{cases} \sum_{i=1}^{0} a_i = 0 \\ \sum_{i=1}^{n} a_i = \left(\sum_{i=1}^{n-1} a_i\right) + a_n, \quad n \geq 1 \end{cases}$
Produtório:	$\begin{cases} \prod_{i=1}^{0} a_i = 1 \\ \prod_{i=1}^{n} a_i = \left(\prod_{i=1}^{n-1} a_i\right) \cdot a_n, \quad n \geq 1 \end{cases}$
União:	$\begin{cases} \bigcup_{i=1}^{0} A_i = \emptyset \\ \bigcup_{i=1}^{n} A_i = \left(\bigcup_{i=1}^{n-1} A_i\right) \cup A_n, \quad n \geq 1 \end{cases}$
Interseção:	$\begin{cases} \bigcap_{i=1}^{0} A_i = U, & \text{(onde } U \text{ é o conjunto universo)} \\ \bigcap_{i=1}^{n} A_i = \left(\bigcap_{i=1}^{n-1} A_i\right) \cap A_n, & n \geq 1 \end{cases}$

◁

Definição recursiva de funções

- $\boxed{\text{Exemplo 10}}$ A sequência de Fibonacci é aquela em que os dois primeiros termos são 0 e 1, e cada termo seguinte é a soma dos dois anteriores:

$$0, \quad 1, \quad 1, \quad 2, \quad 3, \quad 5, \quad 8, \quad 13, \quad \ldots$$

Esta seguência pode ser definida recursivamente como:

$$\begin{cases} f(0) = 0, \\ f(1) = 1, \\ f(n) = f(n-1) + f(n-2), \quad \text{para } n \geq 2. \end{cases}$$

Para calcular $f(5)$ podemos fazer:

$$\begin{aligned} f(2) &= f(1) + f(0) = 1 + 0 = 1, \\ f(3) &= f(2) + f(1) = 1 + 1 = 2, \\ f(4) &= f(3) + f(2) = 2 + 1 = 3, \\ f(5) &= f(4) + f(3) = 3 + 2 = 5. \end{aligned}$$

Definição recursiva de conjuntos e estruturas

- Uma **definição recursiva de um conjunto** tem duas partes:

  Definição recursiva de conjunto:

  Passo base: Especifica-se uma coleção inicial de objetos pertencente ao conjunto.

  Passo recursivo: Especificam-se regras para formar novos elementos a partir dos elementos já pertencentes ao conjunto.

- A definição recursiva de conjuntos também depende da seguinte regra, frequentemente implícita:

  Regra de exclusão: elementos que não podem ser gerados a partir da aplicação do passo base e instâncias do passo indutivo não pertencem ao conjunto.

Definição recursiva de conjuntos e estruturas

- $\boxed{\text{Exemplo 11}}$ Seja o conjunto S definido como:

$$\begin{cases} 3 \in S, \\ \text{se } x \in S \text{ e } y \in S, \text{ então } x + y \in S. \end{cases}$$

Então, é verdade que:

- $6 \in S$? Sim, porque $3 \in S$ e $3 + 3 = 6$,
- $9 \in S$? Sim, porque $3 \in S$ e $6 \in S$ e $3 + 6 = 9$,
- $12 \in S$? Sim, porque $3 \in S$ e $9 \in S$ e $3 + 9 = 12$,
- $7 \in S$? Não, pela regra de exclusão.

O conjunto S é o conjunto dos múltiplos positivos de 3. $\triangleleft$

Definição recursiva de conjuntos e estruturas

- Muitos problemas lidam com **cadeias**, ou **strings**, formadas a partir de um **alfabeto**.
- O conjunto Σ^* de **cadeias** sobre um alfabeto Σ pode ser definido recursivamente como:

  Passo base: $\lambda \in \Sigma^*$ (onde λ representa a cadeia vazia, sem símbolo algum).

  Passo recursivo: Se $w \in \Sigma^*$ e $x \in \Sigma$ então $wx \in \Sigma^*$ (onde wx representa a cadeia formada pelo símbolo x concatenado ao final do prefixo w).

Definição recursiva de conjuntos e estruturas

- $\boxed{\text{Exemplo 12}}$ Seja o alfabeto $\Sigma = \{0, 1\}$. Qual o conjunto de cadeias Σ^* que pode ser formado a partir de Σ?

  Solução.

  Sabemos que:

  - $\lambda \in \Sigma^*$ pelo passo base;
  - $0 \in \Sigma^*$ porque $\lambda \in \Sigma^*$, $0 \in \Sigma$, e podemos juntar $\lambda 0 = 0$;
  - $1 \in \Sigma^*$ porque $\lambda \in \Sigma^*$, $1 \in \Sigma$, e podemos juntar $\lambda 1 = 1$;
  - $00 \in \Sigma^*$ porque $0 \in \Sigma^*$, $0 \in \Sigma$, e podemos juntar 00;
  - $01 \in \Sigma^*$ porque $0 \in \Sigma^*$, $1 \in \Sigma$, e podemos juntar 01;
  - $011 \in \Sigma^*$ porque $01 \in \Sigma^*$, $1 \in \Sigma$, e podemos juntar 011;
  - ...

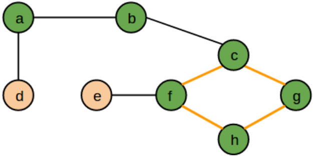
  De fato, Σ^* é o conjunto de todas as cadeias binárias. $\triangleleft$

Definição recursiva de árvores

- Para o próximo exemplo, vamos precisar de alguns conceitos úteis.

- Um grafo $G = (V, E)$ é formado por:
  - um conjunto V de **vértices** ou **nodos**, e
  - um conjunto E de **arestas**, em que cada aresta é um par ordenado (v_i, v_j) indicando que os vértices $v_i, v_j \in V$ estão conectados.
- Um **ciclo** em um grafo é um caminho de arestas consecutivas que começa e termina no mesmo vértice.
- Um **vértice interno** está conectado a pelo menos 2 outros vértices do grafo.
- Uma **folha** é um vértice conectado a no máximo um outro vértice.

- Por exemplo, no grafo abaixo:



  - Vértices são representados por círculos e arestas por linhas conectando vértices.
  - Existe um ciclo começando no vértice c e passando por g, h, f, até voltar em c.
  - Os vértices a, b, c, f, g, h são vértices internos.
  - Os vértice d, e são folhas.

Definição recursiva de árvores

- $\boxed{\text{Exemplo 13}}$ Uma **árvore** é um grafo sem ciclos. Uma **árvore binária completa** é uma árvore em que cada vértice, com exceção das folhas, possui exatamente dois vértices filhos.

  Uma árvore binária completa pode ser definida recursivamente como:

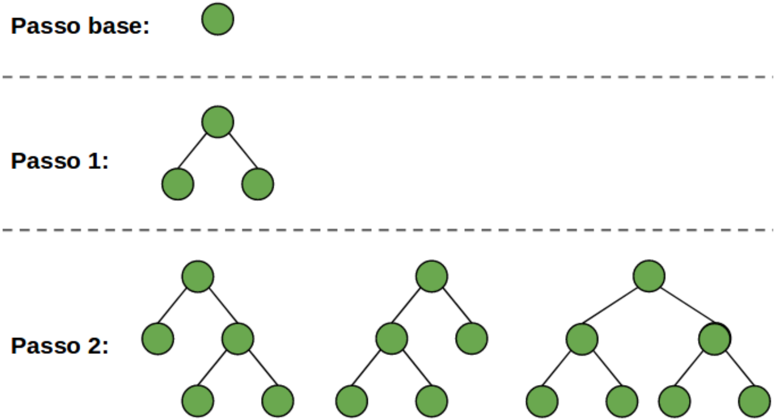
  Passo base: Um vértice isolado é uma árvore binária completa.

  Passo recursivo: Se T_1 e T_2 são árvores binárias completas disjuntas com raízes r_1 e r_2, respectivamente, então pode-se formar uma nova árvore binária completa ao se conectar um vértice r (não presente em T_1 ou T_2, que chamaremos de *raiz*) através de uma aresta a r_1 e outra aresta a r_2.

Definição recursiva de árvores

- Exemplo 13 (Continuação)

Exemplo de construção recursiva de árvores binárias completas:



◁

Indução estrutural

- Se um conjunto tem uma definição recursiva, é possível demonstrar propriedades dos elementos deste conjunto através de indução.
- A **indução estrutural** é uma maneira de mostrar que se:

  1. os elementos iniciais do conjunto (passo base) satisfazem uma certa propriedade, e
  2. as regras de construção de novos elementos (passo indutivo) preservam esta propriedade,

  então todos os elementos do conjunto satisfazem a propriedade.

Indução estrutural

- Uma **demonstração por indução estrutural** tem duas partes:

  Demonstração por indução estrutural:

  Passo base: Mostra-se que a propriedade é verdadeira para todos os elementos especificados no passo base da definição recursiva.

  Passo indutivo: Mostra-se que se a propriedade é verdadeira para cada um dos elementos usados para se construírem novos elementos, então a propriedade também é verdadeira para estes novos elementos.

- A **hipótese de indução** é de que a propriedade é verdadeira para cada um dos elementos usados para construírem-se novos elementos.

Exemplos de uso de indução estrutural

- $\boxed{\text{Exemplo 14}}$ Seja o conjunto S definido recursivamente como:

$$\begin{cases} 3 \in S, \\ x, y \in S \rightarrow x + y \in S. \end{cases}$$

Mostre que todos os elementos de S são divisíveis por 3.

Demonstração. Seja $P(x)$ a proposição *"x é divisível por 3"*.

Passo base: O único elemento da base é 3, e $P(3)$ é verdadeiro porque 3 é divisível por 3 (uma vez que $3 = 3 \cdot 1$ e $1 \in \mathbb{Z}$).

Exemplos de uso de indução estrutural

- $\boxed{\text{Exemplo 14}}$ (Continuação)

Passo indutivo: Assuma que $P(x)$ e $P(y)$ são verdadeiros para dois elementos x e y em A. Ou seja, a I.H. é que os x e y de S são ambos divisíveis por 3.

A regra recursiva diz que podemos usar x e y para incluir o elemento $x + y$ em S, logo que mostrar que $P(x + y)$ também é verdadeiro.
Para isto, note que se x e y são divisíveis por 3, então existem $k', k'' \in \mathbb{N}$ tais que $x = 3k'$ e $y = 3k''$. Nesse caso, podemos derivar:

$$x + y = 3k' + 3k'' = 3(k' + k''),$$

de onde concluímos que $x + y$ também é divisível por 3. Isto conclui o passo indutivo.

Como concluímos com sucesso o passo base e o passo indutivo, a demonstração por indução estrutural está concluída. $\square$

Exemplos de uso de indução estrutural

- Para o próximo exemplo, vamos introduzir mais alguns conceitos sobre árvores binárias completas.
- A **altura** $h(T)$ de uma árvore binária completa T é definida recursivamente como:

$$h(T) = \begin{cases} 0, & \text{se o único vértice da árvore binária} \\ & \text{completa } T \text{ é a própria raiz;} \\ 1 + \max(h(T_1), h(T_2)), & \text{se a árvore binária completa } T \\ & \text{é formada por uma raiz conectada} \\ & \text{a duas sub-árvores } T_1 \text{ e } T_2. \end{cases}$$

Exemplos de uso de indução estrutural

- O **número de vértices** $n(T)$ de uma árvore binária completa T é definido recursivamente como:

$$n(T) = \begin{cases} 1, & \text{se o único vértice da árvore binária} \\ & \text{completa } T \text{ é a própria raiz} \\ 1 + n(T_1) + n(T_2), & \text{se a árvore binária completa } T \\ & \text{é formada por uma raiz conectada} \\ & \text{a duas sub-árvores } T_1 \text{ e } T_2. \end{cases}$$

Exemplos de uso de indução estrutural

- $\boxed{\text{Exemplo 15}}$ Mostre em uma árvore binária completa T, temos

$$n(T) \leq 2^{h(T)+1} - 1 \ .$$

Demonstração. Vamos demonstrar esta desigualdade usando indução estrutural.

Passo base: Para uma árvore binária completa T consistindo apenas num vértice raiz, note que, por definição: $n(T) = 1$ e $h(T) = 0$, logo a desigualdade é satisfeita pois

$$n(T) = 1 \ , \qquad \text{e}$$
$$2^{h(T)+1} - 1 = 2^{0+1} - 1 = 1 \ ,$$

e, portanto,

$$n(T) \leq 2^{h(T)+1} - 1 \ .$$

Exemplos de uso de indução estrutural

- $\boxed{\text{Exemplo 15}}$ (Continuação)

Passo indutivo: A nossa hipótese de indução é que temos

$$n(T_1) \leq 2^{h(T_1)+1} - 1 , \qquad \text{e}$$

$$n(T_2) \leq 2^{h(T_2)+1} - 1$$

sempre que T_1 e T_2 forem árvores binárias completas.

Assuma que T é uma árvore binária completa tendo T_1 e T_2 como sub-árvores imediatas.

As fórmulas recursivas de $n(T)$ e $h(T)$ determinam que

$$n(T) = 1 + n(T_1) + n(T_2) , \qquad \text{e}$$

$$h(T) = 1 + \max(h(T_1), h(T_2)) .$$

Exemplos de uso de indução estrutural

- $\boxed{\text{Exemplo 15}}$ (Continuação)

Assim, podemos computar:

$$
\begin{aligned}
n(T) = & & \text{(def. recursiva de } n(T)\text{)} \\
1 + n(T_1) + n(T_2) \leq & & \text{(hipótese de indução)} \\
1 + (2^{h(T_1)+1} - 1) + (2^{h(T_2)+1} - 1) \leq & & \text{(reorganizando)} \\
(2^{h(T_1)+1} + 2^{h(T_2)+1}) - 1 \leq & & (*) \\
2 \cdot \max(2^{h(T_1)+1}, 2^{h(T_2)+1}) - 1 = & & (\max(2^x, 2^y) = 2^{\max(x,y)}) \\
2 \cdot 2^{\max(h(T_1)+1, h(T_2)+1)} - 1 = & & (\max(x+1, y+1) = \max(x, y)+1) \\
2 \cdot 2^{\max(h(T_1), h(T_2))+1} - 1 = & & \text{(def. recursiva de } h(T)\text{)} \\
2 \cdot 2^{h(T)} - 1 = & & \text{(manipulação algébrica)} \\
2^{h(T)+1} - 1 & &
\end{aligned}
$$

onde o passo (*) vale porque a soma de dois termos é sempre menor ou igual a duas vezes o maior termo. □

Algoritmos Recursivos

Algoritmo recursivos: Introdução

- Às vezes podemos reduzir a solução de um problema com um conjunto particular de valores de entrada para a solução do mesmo problema com valores de entrada menores.

- Quando tal redução pode ser feita, a solução para o problema original pode ser encontrada via uma sequência de reduções, até que o problema tenha sido reduzido a algum caso inicial para o qual a solução é conhecida.

- Veremos que algoritmos que reduzem sucessivamente um problema ao mesmo problema entradas menores são usadas para resolver uma grande variedade de problemas.

  Tais algoritmos são chamados de <u>recursivos</u>.

- Um **algoritmo recursivo** é um algoritmo que resolve um problema reduzindo este problema a uma instância do mesmo problema com entradas menores.

- A seguir vamos ver vários exemplos de algoritmos recursivos.

Exemplos de algoritmos recursivos

- $\boxed{\text{Exemplo 16}}$ Dê um algoritmo recursivo para computar $n!$, onde n é um número inteiro não-negativo.

  Solução. Para construir um algoritmo recursivo que encontre $n!$, onde n é um inteiro não-negativo, podemos nos basear na definição recursiva de $n!$:

$$n! = \begin{cases} n \cdot (n-1)!, & \text{se } n > 0 \\ 1, & \text{se } n = 0 \end{cases}$$

  Esta definição diz que para encontrar $n!$ para um inteiro particular n, podemos usar a etapa recursiva repetidamente vezes, em cada vez substituindo um valor da função fatorial pelo valor da função fatorial no próximo inteiro menor.

  Fazemos isso até atingir o passo base, em que o inteiro a ser computado é 0, e podemos inserir o valor conhecido de $0! = 1$.

Exemplos de algoritmos recursivos

- $\boxed{\text{Exemplo 16}}$ (Continuação)

  Um pseudo-código para um algoritmo recursivo para a função fatorial é o seguinte.

  procedure *factorial*(n: nonnegative integer)
  if $n = 0$ **then return** 1
  else return $n \cdot factorial(n-1)$
  {output is $n!$}

  Exemplo de execução do algoritmo para o valor de entrada $n = 4$:

Função	**Chamada recursiva**	**Valor de retorno**
`factorial(4)`	$4 \cdot$ `factorial(3)`	24
`factorial(3)`	$3 \cdot$ `factorial(2)`	6
`factorial(2)`	$2 \cdot$ `factorial(1)`	2
`factorial(1)`	$1 \cdot$ `factorial(0)`	1
`factorial(0)`	—	1

◁

Exemplos de algoritmos recursivos

- $\boxed{\text{Exemplo 17}}$ Dê um algoritmo recursivo para calcular a^n, onde a é um número real diferente de zero e n é um inteiro não-negativo.

  Solução. Para construir um algoritmo recursivo que encontre a^n, onde a é um real diferente de zero e n é um inteiro não-negativo, podemos nos basear na definição recursiva de a^n:

  $$an = \begin{cases} a \cdot a^{n-1}, & \text{se } n > 0 \\ 1, & \text{se } n = 0 \end{cases}$$

  Esta definição diz que para encontrar a^n para valores particulares de a e n, podemos usar a etapa recursiva repetidamente vezes, em cada vez substituindo um valor da função de exponenciação pelo valor da função de exponenciação com um expoente sendo o próximo inteiro menor.

  Fazemos isso até atingir o passo base, em que o valor a ser computado é a^0, e podemos inserir o valor conhecido de $a^0 = 1$.

Exemplos de algoritmos recursivos

- Exemplo 17 (Continuação)

Um pseudo-código para um algoritmo recursivo para a função de exponenciação é o seguinte.

procedure *power*(a: nonzero real number, n: nonnegative integer)
if $n = 0$ **then return** 1
else return $a \cdot power(a, n - 1)$
{output is a^n}

Exemplo de execução do algoritmo para o valor de entrada $a = 2$, $n = 4$:

Função	Chamada recursiva	Valor de retorno
`power(2,4)`	$2 \cdot$ `power(2,3)`	16
`power(2,3)`	$2 \cdot$ `power(2,2)`	8
`power(2,2)`	$2 \cdot$ `power(2,1)`	4
`power(2,1)`	$2 \cdot$ `power(2,0)`	2
`power(2,0)`	—	1

Exemplos de algoritmos recursivos

- $\boxed{\text{Exemplo 18}}$ Dada uma sequência $L = a_1, a_2, \ldots, a_n$ de de inteiros positivos em ordem crescente e um elemento arbitrário x, uma **pesquisa binária** do elemento x em L funciona assim:

  1. Compare o elemento x a ser pesquisado com o termo do meio da sequência, ou seja, o termo $a_{\lfloor (n+1)/2 \rfloor}$.
  2. Se x for igual a este termo, retorne a localização deste termo no sequência.
  3. Caso contrário:

     a) faça uma nova pesquisa binária na primeira metade da sequência original se x for menor que o termo do meio; ou

     b) faça uma nova pesquisa binária na segunda metade da sequência original se x for maior que o termo do meio; ou

     c) retorne 0 se não há mais elementos a serem pesquisados.

  Expresse a pesquisa binária como um algoritmo recursivo.

Exemplos de algoritmos recursivos

- $\boxed{\text{Exemplo 18}}$ (Continuação)

Solução. Um pseudo-código para um algoritmo recursivo para a função de pesquisa binária é o seguinte.

procedure *binary search*(i, j, x: integers, $1 \leq i \leq j \leq n$)
$m := \lfloor (i + j)/2 \rfloor$
if $x = a_m$ **then**
    return m
else if ($x < a_m$ and $i < m$) **then**
    return *binary search*($i, m - 1, x$)
else if ($x > a_m$ and $j > m$) **then**
    return *binary search*($m + 1, j, x$)
else return 0
{output is location of x in $a_1, a_2, \ldots, a_n$ if it appears; otherwise it is 0}

Exemplos de algoritmos recursivos

- Exemplo 18 (Continuação)

  Exemplo de execução do algoritmo de pesquisa binária para os valores de entrada

  $$i = 1, \qquad j = 12, \qquad x = 20,$$

  na <u>lista ordenada</u>

  $$\begin{array}{cccccc} a_1 = 1, & a_2 = 5, & a_3 = 8, & a_4 = 10, & a_5 = 12, & a_6 = 15, \\ a_7 = 17, & a_8 = 20, & a_9 = 23, & a_{10} = 25, & a_{11} = 28, & a_{12} = 30 : \end{array}$$

Função	**Chamada recursiva**	**Valor de retorno**
`bin_search(1,12,20)`	`bin_search(7,12,20)`	8
`bin_search(7,12,20)`	`bin_search(7,8,20)`	8
`bin_search(7,8,20)`	`bin_search(8,8,20)`	8
`bin_search(8,8,20)`	——	8

Exemplos de algoritmos recursivos

- $\boxed{\text{Exemplo 18}}$ (Continuação)

  Exemplo de execução do algoritmo de pesquisa binária para os valores de entrada

  $$i = 1, \qquad j = 12, \qquad x = 7,$$

  na <u>lista ordenada</u>

  $$\begin{array}{cccccc} a_1 = 1, & a_2 = 5, & a_3 = 8, & a_4 = 10, & a_5 = 12, & a_6 = 15, \\ a_7 = 17, & a_8 = 20, & a_9 = 23, & a_{10} = 25, & a_{11} = 28, & a_{12} = 30: \end{array}$$

Função	**Chamada recursiva**	**Valor de retorno**
`bin_search(1,12,7)`	`bin_search(1,5,7)`	0
`bin_search(1,5,7)`	`bin_search(1,2,7)`	0
`bin_search(1,2,7)`	`bin_search(2,2,7)`	0
`bin_search(2,2,7)`	——	0

◁

Demonstrando a correção de algoritmos recursivos

- Podemos usar a indução matemática para demonstre a correção de algoritmos recursivos.

- $\boxed{\text{Exemplo 19}}$ Demonstre que o algoritmo recursivo que provemos em um exemplo anterior para calcular a exponenciação de um número real com expoente inteiro não-negativo está correto.

  Solução. Vamos usar indução matemática no expoente n.

  Passo base: Se $n = 0$ nosso algoritmo recursivo nos diz que $power(a, 0) = 1$, o que está correto porque $a^0 = 1$ para qualquer número real a.

Demonstrando a correção de algoritmos recursivos

- $\boxed{\text{Exemplo 19}}$ (Continuação)

Passo indutivo: A hipótese de indução é que o algoritmo recursivo computa o valor correto da potência para um inteiro arbitrário, ou seja, que $power(a, k) = a^k$ para qualquer valor real $a \neq 0$ e um inteiro não-negativo arbitrário k.

Temos que mostrar que se a hipótese de indução é verdadeira, então o algoritmo computa a resposta correta para $k + 1$, ou seja, teremos $power(a, k + 1) = a^{k+1}$.

Note que como k é um inteiro positivo, o algoritmo faz $power(a, k + 1) = a \cdot power(a, k)$.

Como pela hipótese de indução temos $power(a, k) = a^k$, concluímos que $power(a, k + 1) = a \cdot a^k = a^{k+1}$, e o algoritmo também está correto para $k + 1$.

$\triangleleft$