4. NOTAÇÕES ITERATIVAS e RECURSIVAS

O objetivo maior desta lição
é servir de preparo para o estudo das demonstrações iterativas e das recursivas, as quais serão abordadas nas próximas lições. Aqui nos limitaremos a apresentar essas duas ideias e trabalhar com a notação envolvida. Nas próximas lições veremos como utilizá-las para resolver problemas e fazer demonstrações.

Embora com muita diferença de conceito e de uso prático, as iterações e as recursões têm em comum o uso da repetição. Observe isso no exemplo da caracterização de uma PA (progressão aritmética), a_0, a_1, a_2, etc., de razão r:

– iteratavamente: $a_n = a_0 + nr$, para todos os $n \geqslant 1$
– recursivamente: $a_n = r + a_{n-1}$, para todos os $n \geqslant 1$.

No que segue, examinaremos em detalhe tanto a ideia geral de iteração como de recursão, e também insistiremos nos cuidados envolvidos na notação envolvida.

1). A ideia de iteração

consiste em definir uma sequência de objetos matemáticos, $\mathbf{O}_1$, $\mathbf{O}_2$, $\mathbf{O}_3$, etc. (os quais podem ser números, funções, matrizes, etc.), mediante a avaliação de uma fórmula do tipo $\mathbf{O}_n = f(n)$, onde f é uma função que caracteriza tal sequência. Consequentemente, a geração sucessiva desses objetos envolve fazer o cálculo *repetido ou iterado:* $\mathbf{O}_1 = f(1)$, $\mathbf{O}_2 = f(2)$, $\mathbf{O}_3 = f(3)$, etc.

Exemplo modelo
No caso da PA acima, temos $f(n) = a_0 + nr$, a qual gera sucessivamente os termos da PA, como segue: $a_1 = f(1) = a_0 + r$, $a_2 = f(2) = a_0 + 2r$, $a_3 = f(3) = a_0 + 3r$, e assim por diante.

Exemplo
Gerar os quatro primeiros elementos da sequência dada por $a_n = n + 2^n$, onde n é um número inteiro positivo.
Na notação acima, $a_n = f(n) = n + 2^n$. Iterando a função, obtemos: $a_1 = f(1) = 1 + 2^1 = 1 + 2 = 3$, $a_2 = f(2) = 2 + 2^2 = 2 + 4 = 6$, $a_3 = f(3) = 3 + 2^3 = 3 + 8 = 11$, e $a_4 = f(4) = 4 + 2^4 = 4 + 16 = 20$.
Note que se fosse pedido determinar o elemento a_{50}, bastaria calcularmos $f(50)$, sendo desnecessário calcular $f(1), f(2), f(3), \ldots, f(49)$. Adiante, veremos a importância dessa observação.

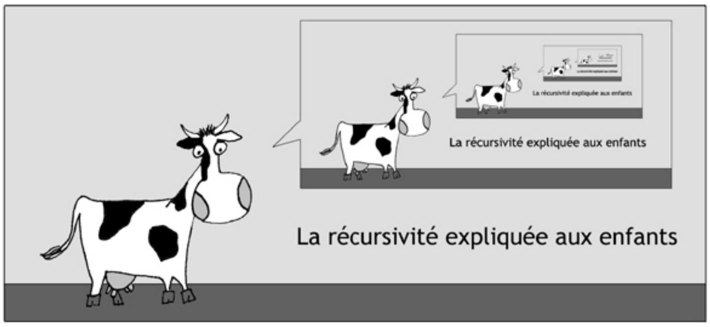
Exemplo
Na Lição 1 negativamos a conjectura que afirmava que os *números de Fermat*, definidos iterativamente pela fórmula $P(n) = 1 + 2^{2^n}$ (onde $n = 0, 1, 2, \ldots$), são primos. Com efeito, vimos que Euler mostrou que $P(5)$ não é primo, pois é fatorável: $P(5) = 1+2^{32} = 4\,294\,967\,297 = 641 \times 6\,700\,417$. Existem outros contraexemplos dessa conjectura? Sim! Por exemplo, $P(6) = 1+2^{64} = 18446744073709551617 = 274177 \times 67280421310721$.

Exemplo
A função $f(n) = \begin{bmatrix} \cos(n\varphi) & -\operatorname{sen}(n\varphi) \\ \operatorname{sen}(n\varphi) & \cos(n\varphi) \end{bmatrix}$, define iterativamente a sequência de matrizes:

$$A_0 = f(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ A_1 = f(1) = \begin{bmatrix} \cos(\varphi) & -\operatorname{sen}(\varphi) \\ \operatorname{sen}(\varphi) & \cos(\varphi) \end{bmatrix},\ A_2 = f(2) = \begin{bmatrix} \cos(2\varphi) & -\operatorname{sen}(2\varphi) \\ \operatorname{sen}(2\varphi) & \cos(2\varphi) \end{bmatrix},\ \text{etc.}$$

Fica como exercício de Trigonometria mostrar que $A_n = (A_1)^n$, para $n = 1, 2, 3, \ldots$.
Mostra-se que A_n produz em cada ponto do plano cartesiano uma rotação anti-horária de ângulo $n\varphi$ e centrada na origem do sistema de eixos.

2). A ideia de recorrência ou recursão



consiste em definir uma sequência de objetos matemáticos, $\mathbf{O}_1$, $\mathbf{O}_2$, $\mathbf{O}_3$, etc. (os quais podem ser números, funções, matrizes, etc.), expressando cada elemento em termos dos que já foram definidos. Para tornar essa ideia mais precisa, consideremos os casos mais comuns de sua aplicação em problemas olímpicos.

– O tipo básico de recursão

Cada elemento é definido se *recorrendo* apenas a seu anterior. Em termos mais precisos: usando uma função f e partindo de um objeto $\mathbf{O}_1$, os demais objetos são gerados recursivamente calculando-se $\mathbf{O}_2 = f(\mathbf{O}_1)$, $\mathbf{O}_3 = f(\mathbf{O}_2)$, $\mathbf{O}_4 = f(\mathbf{O}_3)$, etc. Ou seja, precisamos conhecer o primeiro elemento da sequência (ele será denominado *semente* da recursão) e a recursão, na verdade, só é usada a partir do segundo termo da sequência.

A expressão geral (ou "fórmula") da recursão fica $\mathbf{O}_n = f(\mathbf{O}_{n-1})$, que também pode ser escrita como $\mathbf{O}_{n+1} = f(\mathbf{O}_n)$, a qual é preferível usar na prática pois evita erros de índices e simplifica as fórmulas.

Note que o cálculo sempre inicia com o primeiro objeto depois da semente. Assim se esta for, por exemplo, o objeto $\mathbf{O}_5$, o cálculo iniciará: $\mathbf{O}_6 = f(\mathbf{O}_5)$, $\mathbf{O}_7 = f(\mathbf{O}_6)$, etc. Também note que a determinação recursiva de $\mathbf{O}_{50}$, por exemplo, em uma sequência $\mathbf{O}_1$, $\mathbf{O}_2$, $\mathbf{O}_3$, etc., envolve *obrigatoriamente* o conhecimento/cálculo de $\mathbf{O}_1$, $\mathbf{O}_2$, etc., até o $\mathbf{O}_{49}$, exceto se já conhecermos o $\mathbf{O}_{49}$.

Exemplo modelo

No caso da PA a_0, a_1, a_2, etc., de razão r, temos $f(a_n) = r + a_n$, a qual gera sucessivamente os termos da PA, como segue: $a_1 = f(a_0) = r + a_0$, $a_2 = f(a_1) = r + a_1 = 2r + a_0$, $a_3 = f(a_2) = r + a_2 = 3r + a_0$, e assim por diante.

Exemplo

Gerar os quatro primeiros elementos da sequência dada pela recursão $a_{n+1} = 4 + \frac{a_n}{2}$, *onde* n *é um número inteiro positivo, e o primeiro termo é* 6.

Neste caso, $f(x) = 4 + x/2$, os termos da sequência estão indexados como a_1, a_2, a_3, etc., e o primeiro termo (a semente) da recursão é $a_1 = 6$. Temos, então:

$a_2 = f(a_1) = 4 + \frac{a_1}{2} = 4 + \frac{6}{2} = 7$, $a_3 = f(a_2) = 4 + \frac{a_2}{2} = 4 + \frac{7}{2} = \frac{15}{2}$, $a_4 = f(a_3) = 4 + \frac{a_3}{2} = 4 + \frac{15/2}{2} = \frac{31}{4}$.

Exemplo

Calcular o quarto termo da sequência gerada recursivamente pela semente $u_0 = -2$ *e a fórmula* $u_{n+1} = 3 + (u_n)^2$, *a qual abreviamos como* $u_{n+1} = 3 + u_n^2$.

A partir de u_0, precisamos calcular u_1, u_2, u_3. Isso fica assim:

$$u_1 = 3 + u_0^2 = 3 + (-2)^2 = 3 + 4 = 7, \quad u_2 = 3 + u_1^2 = 3 + 7^2 = 52, \quad u_3 = 3 + u_2^2 = 3 + 52^2 = 3 + 2704 = 2707.$$

Exemplo

Seja a função f *de variável real* x, *dada pela expressão* $f(x) = \frac{1}{1-x}$. *Com isso, definimos a sequência de funções* $f_1, f_2, f_3, \ldots$ *por meio da recursão* $f_{n+1}(x) = f\big(f_n(x)\big)$, *onde a semente é* $f_1 = f$. *Pede-se a expressão analítica dessas funções.*

Temos, sucessivamente:

$f_2(x) = f\big(f_1(x)\big) = \frac{1}{1-f_1(x)} = \frac{1}{1-\frac{1}{1-x}} = \frac{x-1}{x} = 1 - \frac{1}{x}$, para $x \neq 1, 0$;

$f_3(x) = f\big(f_2(x)\big) = \frac{1}{1-f_2(x)} = \frac{1}{1-(1-\frac{1}{x})} = x$, para $x \neq 1, 0$.

A partir do calculado até agora, por que podemos afirmar que a sequência das infinitas funções desta família pode ser escrita como: $f_1, f_2, f_3, f_1, f_2, f_3, f_1, f_2, f_3, \ldots$?

Exemplo

Escrever uma recursão capaz de gerar sequência seguindo o padrão $3,\ 3\sqrt{3},\ 9,\ 9\sqrt{3},\ \ldots$

Tomando $a_1 = 3$, podemos usar $a_{n+1} = a_n \cdot \sqrt{3}$, para $n \geqslant 1$. Confira.

– Recursões de duas ou mais memórias

O tipo de recursão que encontramos acima, o de fórmula $\mathbf{O}_{n+1} = f\,(\mathbf{O}_n)$, é dito ser uma *recursão de uma memória*. Podemos também encontrar recursões de duas ou mais memórias, as fórmulas das quais envolvem funções de duas ou mais variáveis. Um exemplo de fórmula para recursões de duas memórias é: $\mathbf{O}_{n+1} = f\,(\mathbf{O}_n, \mathbf{O}_{n-1})$.

Exemplo

A famosa sequência de Fibonacci é dada por uma recursão de duas memórias: $F_{n+1} = F_n + F_{n-1}$*, e ela é gerada pelas sementes* $F_1 = F_2 = 1$.

Essa recursão é definida pela função soma: $f(F_n, F_{n-1}) = F_n + F_{n-1}$, e *desenrolando* obtemos:
$F_3 = F_2 + F_1 = 1+1 = 2$, $F_4 = F_3 + F_2 = 2+1 = 3$, $F_5 = F_4 + F_3 = 3+2 = 5$, $F_6 = F_5 + F_4 = 5+3 = 8$, etc.

– Existem muitas outras variantes dos tipos encontrados acima.

Seria complicado e infrutífero dar uma fórmula geral para elas, o realmente importante é sermos capazes de perceber o envolvimento da ideia de recorrência. Vejamos alguns exemplos.

Exemplo

Para $n \geqslant 0$*, e* $x_0 = 2$*, definamos* $x_{n+1} = \frac{1+x_n}{n+1}$*. Pede-se o valor de* x_3.

A novidade é que a função caracterizadora da recursão é do tipo $x_{n+1} = f(n, x_n)$. Por outro lado, semelhante ao que ocorreu nos exemplos anteriores, precisaremos conhecer x_0, x_1, x_2 para determinar x_3, e os cálculos iniciam com x_1. Confira:

$$x_1 = \tfrac{1+x_0}{1} = \tfrac{1+2}{1} = 3, \quad x_2 = \tfrac{1+x_1}{2} = \tfrac{1+3}{2} = 2, \quad x_3 = \tfrac{1+x_2}{3} = \tfrac{1+2}{3} = 1.$$

Exemplo (importante)

Na Matemática Computacional, na Engenharia e outras aplicações, tem grande utilidade o que denominamos polinômios de Chebyshev. Esses formam uma sequência T_0, T_1, T_2*, etc., definida recursivamente por* $T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$*, para* $n \geqslant 1$*, a qual tem como sementes iniciais:* $T_0(x) = 1$ *(polinômio constante) e* $T_1(x) = x$*. Pede-se obter a expressão algébrica de* T_2, T_3, T_4, T_5.

Para o terceiro Chebyshev: $T_2(x) = 2xT_1(x) - T_0(x) = 2x \cdot x - 1 = 2x^2 - 1$;
para o quarto: $T_3(x) = 2xT_2(x) - T_1(x) = 2x \cdot (2x^2 - 1) - x = 4x^3 - 3x$.
Fica como exercício verificar que $T_4(x) = 8x^4 - 8x^2 + 1$, e que $T_5(x) = 16x^5 - 20x^3 + 5x$.

3). Comparação entre iteração e recursão

Existem objetos matemáticos que podem ser definidos ou trabalhados tanto iterada como recursivamente. Os exemplos mais conhecidos são os das PA (já visto na introdução) e os das PG. Outro caso é o da fatorial de um inteiro $n \geqslant 1$. Temos

- definição iterativa: $1! = 1$, $2! = 1 \cdot 2$, $3! = 1 \cdot 2 \cdot 3$, $\ldots$, $n! = 1 \cdot 2 \cdot 3 \cdots n$, o que costumamos resumir como $n! = 1 \cdot 2 \cdot 3 \cdots n$, para todos os $n \geqslant 1$;
- definição recursiva: a partir de $1! = 1$, definimos $2! = 2 \cdot 1!$, $3! = 3 \cdot 2!$, etc., $(n+1)! = (n+1) \cdot n!$, o que costumamos resumir como: $1! = 1$ e $(n+1)! = (n+1) \cdot n!$, para todos os $n \geqslant 1$.

No quadro a seguir, fazemos uma breve comparação entre iteração e recursão. Na Computação há enorme literatura sobre isso, pois lá essas duas ideias são muito confundidas.

	iteração	**recursão**
Ideia e fórmula geral	avaliação repetida de uma função: $\mathbf{O}_n = f(n)$, para sucessivos valores de n	avaliação com realimentação dos valores de uma função: $\mathbf{O}_{n+1} = f\,(\mathbf{O}_n)$, para sucessivos valores de n
uso preferencial	na Computação	na Matemática
programando computadores	programas maiores, loop de parada especificada, pequeno uso de memória	programas menores, risco de loop infinito, grande uso de memória.

4). Equivalência entre iteração e recursão

Recorde que acima caracterizamos as PA's e as fatoriais tanto iterativa como recursivamente. Essa dualidade é geral, ou seja: não existe problema intrinsecamente iterativo ou intrinsecamente recursivo. Todo problema iterativo pode ser transformado num recursivo, e vice-versa.

Contudo, essa afirmação é difícil de ser traduzida na forma de um teorema geral e assim não valerá a pena tentarmos fazer isso aqui. Vamos usá-la apenas como um recurso a eventualmente ser aproveitado. Na Matemática, em geral será preferível pensar recursivamente, como veremos nas próximas lições. O contrário ocorre em Informática, pois programas recursivos apesar de pequenos tendem a ser lentos e usar enorme quantidade de memória do computador.

5). Exercícios e problemas

Exercício 1

Seja a sequência de números (a_n), *definida recursivamente por* $a_{n+1} = \frac{a_n}{1+2a_n}$, *tendo como semente* $a_0 = 1/2$. *A partir dela, definimos* $b_n = 1 + \frac{1}{a_n}$. *Pede-se*

- *a). Mostrar que os* b_n *formam uma PA e determine sua razão.*
- *b). Determinar fórmula iterativa tanto para os* a_n *como para os* b_n.

Exercício 2

Achar fórmula iterativa para a sequência dada recursivamente como $x_{n+1} = 3x_n - 1$, *para* $n \geqslant 0$ *e* $x_0 = 1$. *Aplique a fórmula para calcular* x_{100}.

Exercício 3 (importante)

Definamos a sequência (u_n), *pela recursão* $u_{n+2} = \frac{3}{2} \cdot u_{n+1} - \frac{1}{2} \cdot u_n$, *a partir das sementes* $u_0 = 1\,, u_1 = 2$.
Pede-se fórmula iterativa para os u_n.
Sugestão: inicie determinando fórmula iterativa para os termos da sequência $v_n = u_{n+1} - u_n$.

Exercício 4

Considere a família de sequências definidas pela iteração $x_n = n + \operatorname{sen}(an)$, *onde* a *é um parâmetro real. Mostrar que, dependendo do valor atribuído para o parâmetro, pode resultar tanto uma sequência crescente como não crescente (por exemplo oscilante).*

Exercício 5

Na Lição 2, já encontramos as famosas sequências de Syracuse-Collatz. Cada uma delas é definida recursivamente a partir de uma semente inteira $N \geqslant 1$, *e os demais termos são sucessivamente gerados pela seguinte regra: se o último termo conhecido for par, então o próximo é sua metade; caso contrário, o novo termo valerá um somado com o triplo do último. Depois de expressar os termos de tais sequências por meio de fórmula recursiva, refaça o resto do exercício formulado naquela lição.*

Exercício 6

Esta é uma folha de papel A4. Se a dobrarmos ao meio, e ao longo do lado maior, obtemos duas folhas de tamanho A5; se dobrarmos ao meio, e ao longo de seu lado maior, uma folha A5, obtemos duas folhas A6. Todas essas folhas de papel têm a mesma forma, mas o tamanho vai diminuindo. Segundo a norma alemã DIN podemos continuar assim até A10. Também podemos recorrer no sentido inverso, indo de A4 para A3, desta para A2, etc. até chegar à folha de tamanho máximo, a A0.

- *a). Sabendo que as dimensões da A4 são* $21, 21\sqrt{2}$ *cm, deduza as dimensões das folhas A3, A2, A1 e A0. Constate que a área da A0 vale aproximadamente 1 metro quadrado. (A rigor, vale* $84 \times 84\sqrt{2} \simeq 9978\ cm^2$.*)*
- *b). Denotemos por* A_n *a folha An, e definamos como seu comprimento o valor em cm de seu lado maior, e como sua largura o valor em cm de seu lado menor. Pede-se deduzir uma recursão expressando o comprimento e largura de* A_n *em termos do comprimento e largura de* A_{n-1}. *Use esta recursão para reobter as dimensões calculadas no item anterior.*