

Departamento de Matemática
Universidade de Aveiro

Uma pitada de Algoritmia

26/10/2002

Rosália Rodrigues



“It has often been said that a person does not really understand something until he teaches it to someone else. Actually a person does not really understand something until he can teach it to a computer, i.e., express it as an algorithm.”

Donald E. Knuth, American Mathematical Monthly, 1974.



“I hold the opinion that the construction of computer programs is a mathematical activity like the solution of differential equations, that programs can be derived from their specifications through mathematical insight, calculation, and proof, using algebraic laws as simple and elegant as those of elementary arithmetic.”

Sir Charles Antony Richard Hoare.

A Iteração

Muitas vezes,
o **Algoritmo** pode ser construído directamente
a partir da **Formulação Matemática da**
Especificação do Problema.

Especificação: Entrada: $\{n : n \in \mathbb{N}\}$
Saída: $\{soma : soma = \sum_{k=1}^n k\}$

$\{n \in \mathbb{N}\}$
 $k := 1; soma := 1;$
while $k < n$ **do**
 begin $k := k + 1;$
 $soma := soma + k$
 end;
 $\{soma = 1 + 2 + \dots + n\}$

$\{n \in \mathbb{N}\}$
 $soma := 0;$
for $k := 1$ **to** n **do**
 $soma := soma + k;$
 $\{soma = 1 + 2 + \dots + n\}$

Mas estará CERTO?

A condição de paragem do Ciclo:

```
{n ∈ ℕ}
k:= 1;
{k ∈ ℕ }
while k < n do
    begin {k < n} ≡ {k+1 < n+1}
        k:= k+1;
        {k < n+1}

        ...
        {k < n+1}
    end;
{k ≥ n ∧ k < n+1} ⇒ {k = n}
```

O Somatório:

```
{n ∈ ℕ}
k:= 1; soma:= 1;
{k = 1} ∧ {soma=1+2+...+k}
while k < n do
    begin {soma=1+2+...+k} ∧ {k < n}
        k:= k+1;
        {soma=1+2+...+(k-1)} ∧ { k < n+1}
        soma:= soma+k
        {soma=1+2+... +(k-1)+k } ∧ { k < n+1}
    end;
{soma = 1+2+...+k} ∧ {k ≥ n} ∧ {k < n+1}
{soma = 1+2+...+n}
```

A Proposição Lógica

$$\{\text{soma} = 1+2+\dots+k\}$$

chama-se Invariante do Ciclo.

**A noção de Invariante
representa o
Princípio de Indução.**

**Com base na noção de Invariante
de Algoritmos Iterativos
é também possível**

Demonstrar Teoremas.

Teorema: $1 + 2 + 3 + \dots + n = n(n+1) / 2$

Especificação: Entrada: $\{n : n \in \mathbb{N}\}$
Saída: $\{soma : soma = 1 + 2 + 3 + \dots + n \wedge$
 $soma = n(n+1) / 2\}$

O Algoritmo e a Demonstração pretendida:

$\{n \in \mathbb{N}\}$

$k := 1; \text{ soma} := 1;$

$\{k=1\} \wedge \{\text{soma}=1+2+\dots+k = 1\} \wedge \{\text{soma} = k(k+1)/2 = 1\}$

while $k < n$ **do**

begin $\{\text{soma}=1+2+\dots+k\} \wedge \{\text{soma} = k(k+1)/2\} \wedge \{k < n\}$

$k := k+1;$

$\{\text{soma}=1+2+\dots+(k-1)\} \wedge \{\text{soma} = (k-1)k/2\} \wedge \{k < n+1\}$

$\text{soma} := \text{soma} + k$

$\{\text{soma}=1+2+\dots+k\} \wedge \{k < n+1\}$

$\wedge \{\text{soma} = (k-1)k/2 + k = k(k+1)/2\}$

end;

$\{\text{soma} = 1+2+\dots+k\} \wedge \{\text{soma} = k(k+1)/2\} \wedge \{k \geq n\} \wedge \{k < n+1\}$

$\{\text{soma} = 1+2+\dots+n\} \wedge \{\text{soma} = n(n+1)/2\}$

**Utilizando a noção de Invariante,
podemos ainda Deduzir Algoritmos.**

Problema: Calcular os $\underline{n}$ primeiros quadrados perfeitos, um por cada volta de ciclo, utilizando apenas somas.

Especificação: Entrada: $\{n : n \in \mathbb{N}\}$
Saída: $\{soma : soma = 1$
 $= 4$
 $= 9$
 $= 16$
 $= 25$
 $\dots$
 $= n^2\}$

Dedução do Algoritmo : o Ciclo:

$\{n \in \mathbb{N}\}$

$k := 1;$

$\{k \in \mathbb{N}\}$

while $k < n$ **do**

begin $\{k \in \mathbb{N}\} \wedge \{k < n\}$

$k := k + 1;$

$\{k \in \mathbb{N}\} \wedge \{k - 1 < n\}$

?

$\{k \in \mathbb{N}\} \wedge \{k - 1 < n\}$

end;

$\{k \in \mathbb{N}\} \wedge \{k \geq n\} \wedge \{k - 1 < n\}$

$\{k = n\}$

Dedução do Algoritmo : o Somatório:

$\{n \in \mathbb{N}\}$

$k := 1; \text{ soma} := 1;$

$\{k \in \mathbb{N}\} \wedge \{\text{soma} = k^2\}$

while $k < n$ **do**

begin $\{k \in \mathbb{N}\} \wedge \{\text{soma} = k^2\} \wedge \{k < n\}$

$k := k + 1;$

$\{k \in \mathbb{N}\} \wedge \{\text{soma} = (k-1)^2\} \wedge \{k-1 < n\}$

$\{\text{soma} = k^2 - 2k + 1\}$

 ?

$\text{soma} := \text{soma} + \text{termo}$

$\{\text{soma} = k^2\}$

$\{k \in \mathbb{N}\} \wedge \{\text{soma} = k^2\} \wedge \{k-1 < n\}$

end;

$\{k \in \mathbb{N}\} \wedge \{\text{soma} = k^2\} \wedge \{k \geq n\} \wedge \{k-1 < n\}$

$\{\text{soma} = n^2\}$

Dedução do Algoritmo : o Termo do Somatório:

$\{n \in \mathbb{N}\}$

$k := 1; \text{ termo} := 1; \text{ soma} := 1;$

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\}$

while $k < n$ **do**

begin $\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\} \wedge \{k < n\}$

$k := k + 1;$

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2(k-1) - 1\} \wedge \{\text{soma} = (k-1)^2\} \wedge \{k-1 < n\}$

$\{\text{termo} = 2k - 3\} \wedge \{\text{soma} = k^2 - 2k + 1\}$

 ?

$\{\text{termo} = 2k - 1\}$

$\text{soma} := \text{soma} + \text{termo}$

$\{\text{soma} = k^2 - 2k + 1 + 2k - 1 = k^2\}$

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\} \wedge \{k-1 < n\}$

end;

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\} \wedge \{k \geq n\} \wedge \{k-1 < n\}$

$\{\text{soma} = n^2\}$

Algoritmo e sua Demonstração:

$\{n \in \mathbb{N}\}$

$k := 1; \text{ termo} := 1; \text{ soma} := 1;$

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\}$

while $k < n$ **do**

begin $\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\} \wedge \{k < n\}$

$k := k + 1;$

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2(k-1) - 1\} \wedge \{\text{soma} = (k-1)^2\} \wedge \{k-1 < n\}$

$\{\text{termo} = 2k - 3\} \wedge \{\text{soma} = k^2 - 2k + 1\}$

$\text{termo} := \text{termo} + 2;$

$\{\text{termo} = 2k - 3 + 2 = 2k - 1\}$

$\text{soma} := \text{soma} + \text{termo}$

$\{\text{soma} = k^2 - 2k + 1 + 2k + 1 = k^2\}$

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\} \wedge \{k-1 < n\}$

end;

$\{k \in \mathbb{N}\} \wedge \{\text{termo} = 2k - 1\} \wedge \{\text{soma} = k^2\} \wedge \{k \geq n\} \wedge \{k-1 < n\}$

$\{\text{soma} = n^2\}$

Mas afinal, o que faz o algoritmo?

```
k:= 1; termo:= 1; soma:= 1;  
while k < n do  
    begin k:= k+1;  
        termo:= termo + 2;  
        soma:= soma + termo  
    end;
```

k	termo	soma
1	1	1
2	3	1+3 = 4
3	5	1+3+5 = 9
4	7	1+3+5+7 = 16
5	9	1+3+5+7+9 = 25
...		...
n		n ²

... isto é, demonstra que:

$$\sum_{k=1}^n (2k - 1) = n^2$$

pelo Método da Indução.

Temos assim algoritmos capazes de demonstrar que:

$$\blacklozenge \quad 1 + 2 + 3 + \dots + n = n(n+1) / 2$$

$$\blacklozenge \quad 1 + 3 + 5 + 7 + \dots + (2n-1) = n^2$$

Exercícios:

❖ Calcular os n primeiros **cubos** perfeitos, um por cada volta de ciclo, ainda utilizando **apenas somas** .

❖ Por fim, ganhe coragem, junte tudo e demonstre (só com um ciclo e apenas com somas) que:

$$1^3 + 2^3 + 3^3 + \dots + n^3 = n^2 (n+1)^2 / 4 = (1 + 2 + 3 + \dots + n)^2$$

- ❖ E já agora, se temos um Algoritmo para calcular $\mathbf{n^2}$, poderemos alterá-lo por forma a calcular $\sqrt{\mathbf{n}}$?

Claro que, estando em **Aritmética de Inteiros**, só podemos pretender calcular uma aproximação inteira de $\sqrt{n}$.

Especificação: Entrada: $\{n : n \in \mathbb{N}\}$
 Saída: $\{k : k = \lceil \sqrt{n} \rceil\}$

Nota: $k = \lceil x \rceil \Leftrightarrow (k-1) < x \leq k$

$k = \lfloor x \rfloor \Leftrightarrow k \leq x < (k+1)$

O que temos:

$\{n \in \mathbb{N}\}$

k:= 1; termo:= 1; soma:= 1;

while k < n **do**

begin k:= k+1;

 termo:= termo + 2;

 soma:= soma + termo

end;

$\{n \in \mathbb{N}\} \wedge \{soma = k^2\} \wedge \{k-1 < n\} \wedge \{k \geq n\}$

O que pretendemos:

$$\{(k-1) < \sqrt{n}\} \wedge \{\sqrt{n} \leq k\} \equiv \{(k-1)^2 < n\} \wedge \{n \leq k^2\}$$

Ou seja:

$$\{n \in \mathbb{N}\} \wedge \{soma = k^2\} \wedge \{(k-1)^2 < n\} \wedge \{n \leq k^2\}$$

Basta portanto alterar a Condição do Ciclo:

$\{n \in \mathbb{N}\}$

$k := 1; \text{ termo} := 1; \text{ soma} := 1;$

$\{\text{soma} = k^2\}$

while $\text{soma} < n$ **do**

begin $\{\{\text{soma} = k^2\} \wedge \{\text{soma} < n\}\} \Rightarrow \{k^2 < n\}$

$k := k + 1;$

$\{(k-1)^2 < n\}$

$\text{termo} := \text{termo} + 2;$

$\text{soma} := \text{soma} + \text{termo}$

$\{\text{soma} = k^2\} \wedge \{(k-1)^2 < n\}$

end;

$\{n \in \mathbb{N}\} \wedge \{\text{soma} = k^2\} \wedge \{(k-1)^2 < n\} \wedge \{n \leq k^2\}$

$\{n \in \mathbb{N}\} \wedge \{\text{soma} = k^2\} \wedge \{(k-1) < \sqrt{n}\} \wedge \{\sqrt{n} \leq k\}$

$\{n \in \mathbb{N}\} \wedge \{k = \lceil \sqrt{n} \rceil\}$

**Mais alguns Algoritmos que podem ser
construídos directamente
a partir da Formulação Matemática da
Especificação do Problema.**

❖ Calcular o **Valor Máximo** de um conjunto finito:

Especificação: Entrada: $\{ x[1..n] \}$
Saída: $\{ \text{maximo} = \max \{x[1..n]\} \}$

Formulação: $\{ \text{maximo} = \max \{x[1..n]\} \} \equiv$
 $\{ \exists \text{maximo} \in x[1..n] :$
 $\forall i \in [1..n] \Rightarrow x[i] \leq \text{maximo} \}$

❖ **Pesquisar** um elemento num conjunto finito:

Especificação: Entrada: $\{ x, a[1..n] : \text{do mesmo tipo} \}$

Saída: $\{ ? x \in a[1..n] \}$

Formulação: $\{ x \in a[1..n] \} \equiv \{ \exists i \in [1..n] : x = a[i] \}$

Nota: $\{ x \notin a[1..n] \} \equiv \{ \forall i \in [1..n] \Rightarrow x \neq a[i] \}$

❖ Verificar a **Relação de Inclusão** entre dois conjuntos finitos:

Especificação: Entrada: $\{ a[1..m], b[1..n] : \text{do mesmo tipo} \}$
Saída: $\{ ? a[1..m] \subset b[1..n] \}$

Formulação: $\{ a[1..m] \subset b[1..n] \} \equiv$
 $\{ \forall i \in [1..m], \exists j \in [1..n] : a[i] = b[j] \}$

❖ **Ordenar** os elementos de um conjunto finito:

Especificação: Entrada: $\{ x[1..n] \}$
 Saída: $\{ x[1..n] \text{ por ordem não decrescente} \}$

Formulação: $\{ x[1..n] \text{ por ordem não decrescente} \} \equiv$
 $\{ \forall i, j \in [1..n] : i < j \Rightarrow x[i] \leq x[j] \} \equiv$
 $\{ \forall i \in [1..n-1] \Rightarrow x[i] \leq x[i+1] \}$

Nota: A **identificação** de ocorrências de pares: $\{ x[i] > x[i+1] \}$ e
consequente **troca** consiste no Método **BubbleSort**.

Uma questão de Análise Numérica:

Problema: Cálculo de Raízes pelo **Método das Bissecções Sucessivas**

Especificação: Entrada: { Uma Equação na forma $f(x) = 0$,
um intervalo $[a,b] : \exists^1 \text{ raiz} \in [a,b]$,
 $\text{erro} \in \mathbb{R}^+ \}$

Saída: { Um intervalo $[a,b] :$
 $\text{raiz} \in [a,b] \wedge |a-b| < \text{erro} \}$

Facilmente se conclui que:

Invariante do Ciclo: { $\text{raiz} \in [a,b] \}$

Condição do Ciclo: { $|a-b| \geq \text{erro} \}$

Construção do Algoritmo:

$\{f(x) = 0 \wedge \text{raiz} \in [a,b] \wedge \text{erro} \in \mathbb{R}^+\}$

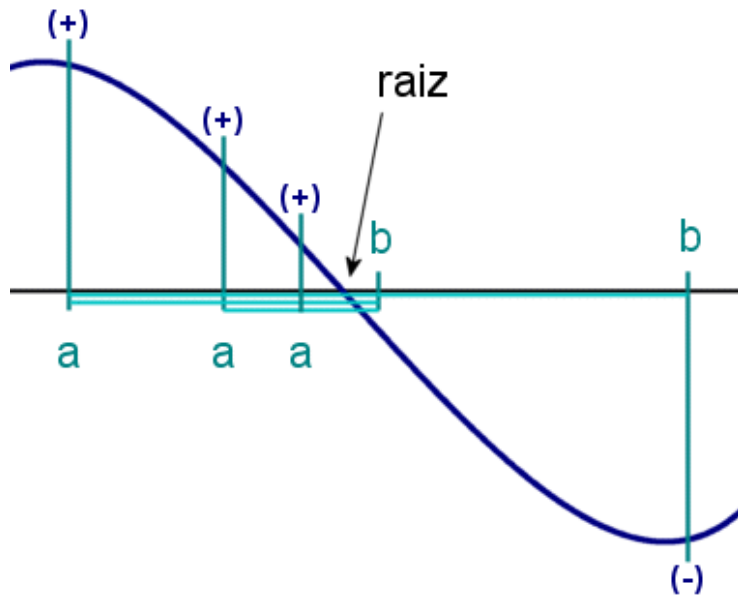
?

$\{\text{raiz} \in [a,b]\}$

```
while abs(a-b) >= erro do  
    begin  $\{\text{raiz} \in [a,b] \wedge |a-b| \geq \text{erro}\}$   
        ?  
         $\{\text{raiz} \in [a,b]\}$   
    end;
```

$\{\text{raiz} \in [a,b] \wedge |a-b| < \text{erro}\}$

O Método das Bissecções Sucessivas:



meio $\leftarrow$ ponto médio de $[a, b]$;
calcular $f(\text{meio})$;

if $raiz \in [\text{meio}, b]$
then $a \leftarrow \text{meio}$
else $b \leftarrow \text{meio}$;

Algoritmo e Prova:

```
{f(x) = 0 ∧ raiz ∈ [a,b] ∧ erro ∈ ℝ+}  
fa:= f(a);  
{raiz ∈ [a,b]}  
while abs(a-b) >= erro do  
    begin {raiz ∈ [a,b] ∧ |a-b| ≥ erro}  
        meio:= (a+b)/2;  
        fm:= f(meio);  
        if (fa > 0) = (fm > 0)  
        then {raiz ∉ [a,meio]} ⇒ {raiz ∈ [meio,b]}  
            a ← meio  
            {raiz ∈ [a,b]}  
        else {raiz ∈ [a,meio]}  
            b ← meio  
            {raiz ∈ [a,b]}  
    end;  
{raiz ∈ [a,b] ∧ |a-b| < erro}
```

Observações:

1. **if** $(fa > 0) = (fm > 0)$
 then $\{raiz \notin [a, meio]\}$
 else $\{raiz \in [a, meio]\}$

f(a)	f(meio)	fa > 0	fm > 0	(fa > 0) = (fm > 0)
+	+	V	V	V
+	-	V	F	F
-	+	F	V	F
-	-	F	F	V

2. **while** $\text{abs}(a-b) \geq \text{erro}$ **do**
 Que garantia temos, quanto à Paragem deste ciclo?

$$\lim_{n \rightarrow \infty} \left\{ \frac{|a - b|}{2^n} \right\} = 0 \iff \forall \varepsilon \in \mathbb{R}^+, \exists N : \forall n \geq N \Rightarrow \frac{|a - b|}{2^n} < \varepsilon$$