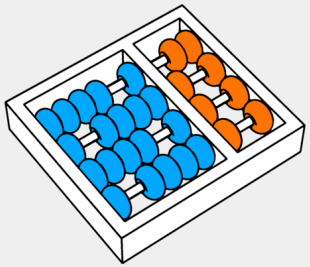
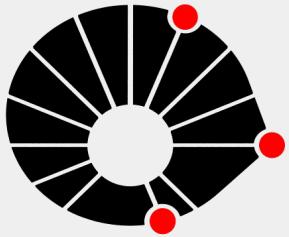
MC358 - Fundamentos matemáticos da computação

Prof. Dr. Hilder Vitor Lima Pereira



Instituto de computação



UNICAMP

06 de setembro de 2023

1. Corretude de algoritmos: continuação

2. Perguntas, observações, comentários?

Corretude de algoritmos: continuação

Divisão euclidiana

Na aula passada, vimos um algoritmo que utiliza apenas adição e subtração para calcular divisão inteira com resto.
Isto é, dados $a, b \in \mathbb{Z}$, temos que $(q, r) = \texttt{DivEuc}(\texttt{a}, \texttt{b})$ satisfaz

$$a = b \cdot q + r \quad \text{e} \quad 0 \leq r < b$$

.
Além disso, sabemos que esse par (q, r) é único.

MDC

O máximo divisor comum de dois inteiros a e b, denotado por $\texttt{mdc}(a, b)$, é definido como o maior inteiro d que divide ambos a e b.

Seja $D_x = \{m \in \mathbb{Z} : m \mid x\}$ para qualquer $x \in \mathbb{Z}$.
Seja $D_{a,b} = D_a \cap D_b = \{m \in \mathbb{Z} : (m \mid a) \wedge (m \mid b)\}$.
Então $\texttt{mdc}(a, b) = \max D_{a,b}$.

MDC

O máximo divisor comum de dois inteiros a e b, denotado por $\texttt{mdc}(a, b)$, é definido como o maior inteiro d que divide ambos a e b.

Seja $D_x = \{m \in \mathbb{Z} : m \mid x\}$ para qualquer $x \in \mathbb{Z}$.
Seja $D_{a,b} = D_a \cap D_b = \{m \in \mathbb{Z} : (m \mid a) \wedge (m \mid b)\}$.
Então $\texttt{mdc}(a, b) = \max D_{a,b}$.

Por exemplo:

(a, b)	$D_{a,b}$	$\texttt{mdc}(a, b)$
$(4, 10)$	$\{\pm 1, \pm 2\}$	2
$(4, 20)$	$\{\pm 1, \pm 2, \pm 4\}$	4
$(45, 90)$	$\{\pm 1, \pm 3, \pm 5, \pm 9, \pm 15, \pm 45\}$	45

Algumas propriedades básicas do MDC

- $\texttt{mdc}(a, b) = \texttt{mdc}(b, a)$ (segue da comutatividade de conectivo $\wedge$)
- $\texttt{mdc}(a, 0) = a$.
- Se $a \mid b$, então $\texttt{mdc}(a, b) = a$.
- Se p é primo, então
  - $\texttt{mdc}(p, a) = p$ se, e somente se, $p \mid a$.
  - $\texttt{mdc}(p, a) = 1 \Leftrightarrow p \nmid a$.
- Definição: dizemos que a e b são coprimos se $\texttt{mdc}(a, b) = 1$.

MDC e o resto da divisão

Para calcular $\texttt{mdc}(a, b)$, usamos a seguinte propriedade:

Teorema

Sejam $a, b \in \mathbb{N}$ tais que $a \geq b > 0$. Seja r o resto da divisão de a por b. Então, $\texttt{mdc}(a, b) = \texttt{mdc}(b, r)$.

MDC e algoritmo de Euclides

Algorithm: `AlgoEuclides`

Input: $a, b \in \mathbb{N}$ com $a \geq b > 0$.

Output: $\texttt{mdc}(a, b)$

```
1 x = a
2 y = b
  ▷ Esta linha marca o caso base
3 while (0 ≠ x mod y) do
      ▷ Esta linha marca o início da iteração
4     r = x mod y
5     x = y
6     y = r
      ▷ Esta linha marca o final da iteração
7 return y
```

MDC e algoritmo de Euclides

Algorithm: `AlgoEuclides`

Input: $a, b \in \mathbb{N}$ com $a \geq b > 0$.
Output: $\texttt{mdc}(a, b)$

```
1 x = a
2 y = b
  ▷ Esta linha marca o caso base
3 while (0 ≠ x mod y) do
      ▷ Esta linha marca o início da iteração
4     r = x mod y
5     x = y
6     y = r
      ▷ Esta linha marca o final da iteração
7 return y
```

- Terminação: A cada duas iterações, ou o algoritmo termina, ou ambos x e y reduzem de tamanho.

MDC e algoritmo de Euclides

Algorithm: `AlgoEuclides`

Input: $a, b \in \mathbb{N}$ com $a \geq b > 0$.

Output: $\texttt{mdc}(a, b)$

```
1 x = a
2 y = b
  ▷ Esta linha marca o caso base
3 while (0 ≠ x mod y) do
      ▷ Esta linha marca o início da iteração
4     r = x mod y
5     x = y
6     y = r
      ▷ Esta linha marca o final da iteração
7 return y
```

- Terminação: A cada duas iterações, ou o algoritmo termina, ou ambos x e y reduzem de tamanho.
- Caso base: $\texttt{mdc}(x, y) = \texttt{mdc}(a, b)$

MDC e algoritmo de Euclides

```
Algorithm: AlgoEuclides
Input:  a, b ∈ ℕ com a ≥ b > 0.
Output: mdc(a, b)
1 x = a
2 y = b
  ▷ Esta linha marca o caso base
3 while (0 ≠ x mod y) do
      ▷ Esta linha marca o início da iteração
4     r = x mod y
5     x = y
6     y = r
      ▷ Esta linha marca o final da iteração
7 return y
```

- Terminação: A cada duas iterações, ou o algoritmo termina, ou ambos x e y reduzem de tamanho.
- Caso base: $\texttt{mdc}(x, y) = \texttt{mdc}(a, b)$
- Invariante: $\texttt{mdc}(x, y) = \texttt{mdc}(a, b)$

Teorema (Identidade de Bézout)

Para todo $a, b \in \mathbb{Z}$, existem $u, v \in \mathbb{Z}$ tais que

$$\texttt{mdc}(a, b) = u \cdot a + v \cdot b$$

onde assumimos $\texttt{mdc}(0, 0) = 0$.

u e v são chamados coeficientes de Bézout* e eles não são únicos.

Exemplo:

- $\texttt{mdc}(2, 15) = 1 = (-7) \cdot 2 + 1 \cdot 15$
- $\texttt{mdc}(20, 12) = 4 = (-1) \cdot 20 + 2 \cdot 12$
- $\texttt{mdc}(3, 15) = 3 = 11 \cdot 3 + (-2) \cdot 15 = 1 \cdot 3 + 0 \cdot 15.$

*Étienne Bézout (1730–1783), matemático francês, provou essa igualdade para polinômios. A identidade para inteiros já era conhecida, mas acabou sendo frequentemente referênciada pelo seu nome...

Uma aplicação modesta da identidade de Bézout

Na aula anterior, dissemos que se p é primo e $p \mid (ab)$, então $p \mid a$ ou $p \mid b$, mas não apresentamos uma prova...
Como podemos provar essa afirmação?

O algoritmo de Euclides estendido

Algorithm: `AlgoEuclidesEstendido`

Input: $a, b \in \mathbb{N}$ com $a \geq b \geq 0$.

Output: $(d, u, v) \in \mathbb{Z}^3$ tais que $d = \texttt{mdc}(a, b)$ e $d = u \cdot a + v \cdot b$

```
1 if 0 == b then
2     return (a, 1, 0)
3 q, r = DivEuc(a, b)
4 (d, u, v) = AlgoEuclidesEstendido(b, r)
5 return (d, v, u − q · v)
```

O algoritmo de Euclides estendido

Algorithm: `AlgoEuclidesEstendido`

Input: $a, b \in \mathbb{N}$ com $a \geq b \geq 0$.

Output: $(d, u, v) \in \mathbb{Z}^3$ tais que $d = \texttt{mdc}(a, b)$ e $d = u \cdot a + v \cdot b$

```
1 if 0 == b then
2     return (a, 1, 0)
3 q, r = DivEuc(a, b)
4 (d, u, v) = AlgoEuclidesEstendido(b, r)
5 return (d, v, u − q · v)
```

- Terminação: Lembrem-se da divisão euclidiana... No pior caso, temos $r \leq b - 1$...

O algoritmo de Euclides estendido

Algorithm: `AlgoEuclidesEstendido`

Input: $a, b \in \mathbb{N}$ com $a \geq b \geq 0$.

Output: $(d, u, v) \in \mathbb{Z}^3$ tais que $d = \texttt{mdc}(a, b)$ e $d = u \cdot a + v \cdot b$

```
1 if 0 == b then
2     return (a, 1, 0)
3 q, r = DivEuc(a, b)
4 (d, u, v) = AlgoEuclidesEstendido(b, r)
5 return (d, v, u − q · v)
```

- Terminação: Lembrem-se da divisão euclidiana... No pior caso, temos $r \leq b - 1$...
- Caso base: Se $b = 0$, então $\texttt{mdc}(a, b) = a = 1 \cdot a + 0 \cdot b$.

O algoritmo de Euclides estendido

Algorithm: `AlgoEuclidesEstendido`

Input: $a, b \in \mathbb{N}$ com $a \geq b \geq 0$.

Output: $(d, u, v) \in \mathbb{Z}^3$ tais que $d = \texttt{mdc}(a, b)$ e $d = u \cdot a + v \cdot b$

```
1 if 0 == b then
2     return (a, 1, 0)
3 q, r = DivEuc(a, b)
4 (d, u, v) = AlgoEuclidesEstendido(b, r)
5 return (d, v, u − q · v)
```

- Terminação: Lembrem-se da divisão euclidiana... No pior caso, temos $r \leq b - 1$...
- Caso base: Se $b = 0$, então $\texttt{mdc}(a, b) = a = 1 \cdot a + 0 \cdot b$.
- Use indução forte: existe $k - 1 \geq 0$ tal que $\texttt{AlgoEuclidesEstendido}(a, r)$ funciona para $0 \leq r \leq k - 1$. Prove que $\texttt{AlgoEuclidesEstendido}(a, k)$ funciona.

O algoritmo de Euclides estendido

Algorithm: `AlgoEuclidesEstendido`

Input: $a, b \in \mathbb{N}$ com $a \geq b \geq 0$.

Output: $(d, u, v) \in \mathbb{Z}^3$ tais que $d = \texttt{mdc}(a, b)$ e $d = u \cdot a + v \cdot b$

```
1 if 0 == b then
2     return (a, 1, 0)
3 q, r = DivEuc(a, b)
4 (d, u, v) = AlgoEuclidesEstendido(b, r)
5 return (d, v, u − q · v)
```

- Terminação: Lembrem-se da divisão euclidiana... No pior caso, temos $r \leq b - 1$...
- Caso base: Se $b = 0$, então $\texttt{mdc}(a, b) = a = 1 \cdot a + 0 \cdot b$.
- Use indução forte: existe $k - 1 \geq 0$ tal que $\texttt{AlgoEuclidesEstendido}(a, r)$ funciona para $0 \leq r \leq k - 1$. Prove que $\texttt{AlgoEuclidesEstendido}(a, k)$ funciona.

O algoritmo de Euclides estendido

Algorithm: `AlgoEuclidesEstendido`

Input: $a, b \in \mathbb{N}$ com $a \geq b \geq 0$.
Output: $(d, u, v) \in \mathbb{Z}^3$ tais que $d = \texttt{mdc}(a, b)$ e $d = u \cdot a + v \cdot b$

```
1 if 0 == b then
2     return (a, 1, 0)
3 q, r = DivEuc(a, b)
4 (d, u, v) = AlgoEuclidesEstendido(b, r)
5 return (d, v, u − q · v)
```

- Terminação: Lembrem-se da divisão euclidiana... No pior caso, temos $r \leq b - 1$...
- Caso base: Se $b = 0$, então $\texttt{mdc}(a, b) = a = 1 \cdot a + 0 \cdot b$.
- Use indução forte: existe $k - 1 \geq 0$ tal que $\texttt{AlgoEuclidesEstendido}(a, r)$ funciona para $0 \leq r \leq k - 1$. Prove que $\texttt{AlgoEuclidesEstendido}(a, k)$ funciona.

Note que esse algoritmo (com sua prova de corretude) nos dá uma prova construtiva da existência dos coeficientes de Bézout para $a \geq b \geq 0$!
E é fácil estender esse algoritmo para quaisquer $a, b \in \mathbb{Z}$!

Perguntas, observações, comentários?