6

# Терминация

Дали алгоритъмът приключва?

def: Фундирана релация в X.

частична наредба в множество X, такава че всяко непразно подмножество на X има мин. елемент

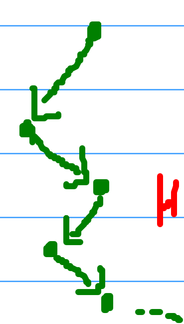
$$(\forall S \subseteq X)\left[S \neq \emptyset \to (\exists m \in S)(\forall s \in S)\neg(s R m)\right]$$

$\langle \mathbb{N}, \leq \rangle$ ✓

$\langle \mathbb{Z}, \leq \rangle$ ✗

- Във фундирано м-во няма "безкрайни спускания"

$a > b > c > d \ldots$



Не!

• evenCount

Терминацията е очевидна!

• taskMystery

t е unsigned

$\langle \mathbb{N}, \leq \rangle$ фундирано

Ще покажем, че алгоритъмът терминира

Допускаме противното.

Нека тази безкрайна редица са последователните с-ти на $t$. защото допуснахме, че не терминира

1-ва итерация 2-ра итерация 3-та

$$t', t'', t''', t'''' \ldots$$

Да разгледаме две последователни стойности на $t$.

$$t^{k} \text{ и } t^{k+1}$$

1сл. $t^{k+1} = t^{k}/2$ $\quad \cdot\ t^{k+1} < t^{k}$

2сл. $t^{k+1} = t^{k} - 1$ $\cdot\ t^{k+1} < t^{k}$

$\Rightarrow$

$t > t' > t'' > t''' > t'''' \ldots$

Допуснахме, че алгоритъмът не терминира

$\Rightarrow$ редицата е безкрайно спускане

но $\langle \mathbb{N}, \leq \rangle$ е фундирано

т.е. няма безкрайни спускания.

Противоречие ↯

$\Rightarrow$ taskMystery терминира

Зад Доказателство за коректност на GCD

Твърдения:

- 1) $НОД(x, y) = НОД(y, x)$
- 2) $НОД(x, 0) = НОД(0, x) = x$
- 3) $y \geq x \Rightarrow НОД(x, y) = НОД(x, y - x)$

1) и 2) - очевидни

Д-во на 3)

Нека $y \geq x$ и НОД$(x, y) = t$

$\Rightarrow t \mid x$ и $t \mid y \Rightarrow t \mid y - x$

Допускаме, че $t$ не е НОД$(x, y-x)$

$\Rightarrow \exists r \, (r > t)\,(r \mid x$ и $r \mid y - x)$

$x = a.r$ $\qquad$ $y - x = b.r$

$\Rightarrow y = (a+b).r$

$\Rightarrow r \mid y$

$\Rightarrow r \mid x$ и $r \mid y$ и $r > t$ противоречие

$\Rightarrow$ НОД$(x, y-x) = t$

**Инварианта**: За всяка проверка за край на цикъла **НОД**(a,b) = **НОД**(x,y)

При първата проверка за край:

x = a и y = b **НОД**(a,b) = **НОД**(x,y) **OK!**

**Поддръжка**:

Допускаме, че инвариантата е изпълнена за някоя проверка за край, **която не е последна**

**НОД**(a,b) = **НОД**(x,y)

**1 сл** y >= x.

Нека с y' означим новата стойност на y. **y'** = y - x

Вярно ли е, че **НОД**(a,b) = **НОД**(x,**y'**) ?

**НОД**(x,**y'**) = **НОД**(x,y-x) (от **тв 3**) = **НОД**(x,y) (от **допускането**) = **НОД**(a,b) **OK!**

**2 сл** y < x.

Нека с **x'** и **y'** означим новите стойности на x и y. **x'** = y и **y'** = x;

Вярно ли е, че **НОД**(a, b) = **НОД**(**x'**,**y'**) ?

**НОД**(**x'**, **y'**) = **НОД**(y, x) (от **тв 1**) = **НОД**(x,y) (от **допускането**) = **НОД**(a,b) OK!

**Терминация**:

Цикълът приключва при x = 0.

**НОД**(a,b) (от **инвариантата**) = **НОД**(x,y) = **НОД**(0, y) (от **тв2**) = y

На ред 17 връщаме точно y! **OK!**

Дали GCD терминира ?

$\langle \mathbb{N}, \leq \rangle$ фундирано

Ще покажем, че алгоритъмът терминира

Допускаме противното:

Нека тази безкрайна редица са последователните с-ти на $X$

$$X', X'', X''', X'''' \dots$$

Да разгледаме две последователни стойности на $x$.

$$x^{k} \qquad x^{k+1}$$

Ако $y \geq x$, то $x^{k+1} = x$ $(\cancel{x^{k} > x^{k+1}})$

Не можем да го докажем така

Друг начин

$$(\mathbb{N} \times \mathbb{N}, \leq)$$

$$(a,b) \leq (a',b') \Longleftrightarrow (a \leq a') \vee (a = a' \wedge b \leq b')$$

Пример: $(\underbrace{3,7) \le (4}_{\le},8)$

док. за домашно

$$(\underbrace{3,\overbrace{8) \le (3}^{=},9}_{\le})$$

$\le$ е **фундирана релация** в $\mathbb{N}\times\mathbb{N}$

$\Rightarrow$ Няма безкрайни спускания

Нека тази **безкрайна** редица са последователните с-ти на $x$ и $y$

$$(x', y'),\ (x'', y''),\ (x''', y''')\ \dots$$

Да разгледаме две последователни стойности на $x$ и $y$.

1 сл.

$(x \neq 0)$ $y \geq x$

$$(x^k, y^k) \quad (x^{k+1}, y^{k+1}) = (x^k, y^k - x^k)$$

$$(x^k, y^k) > (x^k, \underbrace{y^k - x^k}_{>})$$

2 сл. ($x \neq 0$)  $y^k < x^k$  $(x^k, y^k) = \overbrace{(y^k, x^k)}^{(x^{k+1}, y^{k+1})}$

$$(x^k, y^k) > \underbrace{(y^k, x^k)}_{(x^{k+1}, y^{k+1})}$$

$x^k > y^k$

$$\Rightarrow (x', y') > (x'', y'') > (x''', y''') \ \ldots$$

Допуснахме, че алгоритъмът не терминира

$\Rightarrow$ редицата е безкрайно спускане

Но $\langle \mathbb{N} \times \mathbb{N}, \leq \rangle$ е фундирано

Противоречие!

$\Rightarrow$ GCD терминира

Зад Коректност на reverse

Ако n не завършва на 0, то reverse връща n обърнато

- $\alpha_1 \alpha_2 \ldots \alpha_n \% 10 = \alpha_n$
- $\beta_1 \beta_2 \ldots \beta_n * 10 + \alpha_i = \beta_1 \ldots \beta_n \alpha_i$
- $\alpha_1 \ldots \alpha_{n-1} \alpha_n / 10 = \alpha_1 \ldots \alpha_{n-1}$

$n = 1234$

| temp | result |
|---|---|
| 1234 | 0 |
| 123 | 4 |
| 12 | 43 |
| 1 | 432 |
| 0 | 4321 |

$$x \circ y = \begin{cases} y & x = 0 \\ x & y = 0 \\ xy & x \neq 0 \wedge y \neq 0 \end{cases}$$

конкатенираме цифрите.

За всяка проверка за край на цикъла е изпълнено:

**n = temp o (result)^rev**

**База**: При първата проверка за край:

temp = n , result = 0.

Вярно ли е, че temp o (result)^rev = n?

n o 0^rev = n o 0 (от дефиницията на операцията) = n **OK**!

**Поддръжка**: Допускаме, че инвариантата е изпълнена за някоя проверка за край, която **не е последна**. Следователно **temp > 0**.

Допуснали сме: **n = temp o (result)^rev**

temp = a_1 a_2 ... a_k-1 a_k (a_i in {0...9} )

result = b_1 b_2 ...b_r-1 b_r (b_i in {0...9} )

Тогава от допускането: : **n = temp o (result)^rev** = a_1 a_2 ... a_k-1 a_k b_r b_r-1 ... b_1

Нека с **temp'** и **result'** означим новите стойности на **temp** и **result**.

**temp'** = a_1 a_2 ... a_k-1

**result'** = b_1 b_2 ... b_r a_k

**result' != 0** (съдържа сме поне 1 цифра)

Вярно ли е, че n = **temp'** o (**result'**)^rev ??

**temp'** o (**result'**)^rev = a_1 a_2 ... a_k-1 o (b_1 b_2 ... b_r a_k)^rev =

= a_1 a_2 ... a_k-1 a_k b_r b_r-1 ... b_1 = (от допускането) = n **OK!**

**Терминация**:

Последната проверка е при temp = 0.

От инвариантата знаем:

n = **temp** o (**result**)^rev

Но **temp** = 0. От дефиницията на операцията следва че:

n = (**result**)^rev , но тогава: n^rev = **result**

На ред 19 връщаме точно **result**. **OK!**

Терминацията е очевидна!