

6

Терминация

Дали алгоритъмт приключва?

def: Функционална релация в X .

частична наредба в множество X ,

ТАКАВА ЧЕ **Всяко непразно**

подмножество на X има **мин. елемент**

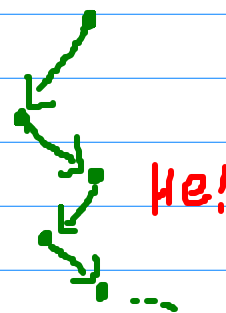
$$(\forall S \subseteq X) [S \neq \emptyset \rightarrow (\exists m \in S)(\forall s \in S) \neg (s R m)]$$

$$\langle \mathbb{N}, \leq \rangle \quad \checkmark$$

$$\langle \mathbb{Z}, \leq \rangle \quad \times$$

- Във функционално м-во **няма** "безкрайни спускания"

$$a > b > c > d \dots$$



- 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 ДОПУСКАМЕ,**
ЧЕ НЕ ТЕРМИНИРА

1-ва итерация → t' , 2-ра итерация → t'' , 3-та итерация → t''' , t'''' ...

Да разгледаме две последователни стойности на t .

t^k и t^{k+1}

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

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

\Rightarrow

$$t > t' > t'' > t''' > t^{(4)} \dots$$

Допуснахме, че алгоритъмът не терминира

\Rightarrow редицата е безкрайно спускане

Но $\langle |N| \leq \rangle$ е фундирано

т.е. няма безкрайни спускания.

Противоречие \downarrow

\Rightarrow taskMystery терминира

зау Доказателство за коректност на GCD

Твърдения:

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

1) и 2) - очевидно

Д-во на 3)

Нека $y \geq x$ и $\text{НОД}(x, y) = t$
 $\Rightarrow t \mid x$ и $t \mid y \Rightarrow t \mid y - x$

Допускаме, че t **не е** $\text{НОД}(x, y)$

$$\Rightarrow \exists r (r > t) (r \mid x \text{ и } r \mid y - x)$$

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

$$\Rightarrow r \mid y$$

$$\Rightarrow r \mid x \text{ и } r \mid y \text{ и } r > t \text{ противоречие}$$

$$\Rightarrow \text{НОД}(x, y-x) = t$$

Инварианта: За всяка проверка за край на цикъла $\text{НОД}(a,b) = \text{НОД}(x,y)$

При първата проверка за край:

$x = a$ и $y = b$ $\text{НОД}(a,b) = \text{НОД}(x,y)$ **ОК!**

Поддръжка:

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

$\text{НОД}(a,b) = \text{НОД}(x,y)$

1 сл $y \geq x$.

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

Вярно ли е, че $\text{НОД}(a,b) = \text{НОД}(x,y')$?

$\text{НОД}(x,y') = \text{НОД}(x,y-x)$ (от **тв 3**) = $\text{НОД}(x,y)$ (от **допускането**) = $\text{НОД}(a,b)$ **ОК!**

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 $\text{НОД}(a,b) = \text{НОД}(x',y')$?

$\text{НОД}(x',y') = \text{НОД}(y,x)$ (от **тв 1**) = $\text{НОД}(x,y)$ (от **допускането**) = $\text{НОД}(a,b)$ **ОК!**

Терминация:

Цикълът приключва при $x = 0$.

$\text{НОД}(a,b)$ (от **инвариантата**) = $\text{НОД}(x,y) = \text{НОД}(0,y)$ (от **тв2**) = y

На ред 17 връщаме точно y ! **ОК!**

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 .

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

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

Не можем да го докажем така

Друг начин

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

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

Пример: $(3, 7) \leq (4, 8)$

док. за
домашно

$$(3, 8) \leq (3, 9)$$

\leq е фундирана релация в $\mathbb{N} \times \mathbb{N}$

\Rightarrow Няма безкрайни спускащи

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 сп.

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

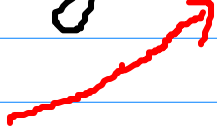
$$(x^k, y^k) \quad (x^{k+1}, y^{k+1}) = (x^k, y^k - x^k)$$

$$(x^k, y^k) > (x^k, y^k - x^k)$$

2 сл.
($x \neq 0$)

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

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

$x^k > y^k$ 

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

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

3.9 Коректност на `reverse`

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

- $d_1 d_2 \dots d_n \neq 10 = d_n$
- $\beta_1 \beta_2 \dots \beta_n \neq 10 + d_i = \beta_1 \dots \beta_n d_i$
- $d_1 \dots d_{n-1} d_n / 10 = d_1 \dots d_{n-1}$

$n = 1234$

`temp`

1234

123

12

1

0

`result`

0

4

43

432

4321

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

↑ конкатениране
цифрите.

За всяка проверка за край на цикъла е изпълнено:

$$n = \text{temp} \circ (\text{result})^{\text{rev}}$$

База: При първата проверка за край:

$$\text{temp} = n, \text{result} = 0.$$

Вярно ли е, че $\text{temp} \circ (\text{result})^{\text{rev}} = n$?

$$n \circ 0^{\text{rev}} = n \circ 0 \text{ (от дефиницията на операцията)} = n \quad \text{OK!}$$

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

$$\text{Допуснали сме: } n = \text{temp} \circ (\text{result})^{\text{rev}}$$

$$\text{temp} = a_1 a_2 \dots a_{k-1} a_k \quad (a_i \in \{0 \dots 9\})$$

$$\text{result} = b_1 b_2 \dots b_{r-1} b_r \quad (b_i \in \{0 \dots 9\})$$

$$\text{Тогава от допускането: } n = \text{temp} \circ (\text{result})^{\text{rev}} = a_1 a_2 \dots a_{k-1} a_k b_r b_{r-1} \dots b_1$$

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

$$\text{temp}' = a_1 a_2 \dots a_{k-1}$$

$$\text{result}' = b_1 b_2 \dots b_r a_k$$

$\text{result}' \neq 0$ (съдържа сме поне 1 цифра)

Вярно ли е, че $n = \text{temp}' \circ (\text{result}')^{\text{rev}}$??

$$\begin{aligned} \text{temp}' \circ (\text{result}')^{\text{rev}} &= a_1 a_2 \dots a_{k-1} \circ (b_1 b_2 \dots b_r a_k)^{\text{rev}} = \\ &= a_1 a_2 \dots a_{k-1} a_k b_r b_{r-1} \dots b_1 = (\text{от допускането}) = n \quad \text{OK!} \end{aligned}$$

Терминация:

Последната проверка е при $\text{temp} = 0$.

От инвариантата знаем:

$$n = \text{temp} \circ (\text{result})^{\text{rev}}$$

Но $\text{temp} = 0$. От дефиницията на операцията следва че:

$$n = (\text{result})^{\text{rev}}, \text{ но тогава: } n^{\text{rev}} = \text{result}$$

На ред 19 връщаме точно result . OK!

Терминацията е очевидна!