

Намиране на НОД на две цели положителни числа

Задача: Дадени са целите числа A и B , $A > B > 0$. Да се намери най-голямото цяло число, което дели без остатък както A , така и B .

Решение:

Ако не знаем алгоритъма на Евклид, бихме могли да решим задачата по следния начин. Проверяваме дали B не е общ делител. Ако не е, проверяваме за целите числа от $B-1$ до 1 дали не са общи делители. Първият намерен общ делител ще бъде НОД. Тази процедура обаче изисква много пресмятания. Алгоритъмът на Евклид с деление е по-бърз и се състои в следното.

За A и B е в сила представянето $A = M_1 \cdot B + N_1$, $0 \leq N_1 < B$. Продължаваме с представяне на B и N_1 , N_1 и N_2 и т.н. докато получим $N_k = 0$.

$$\begin{aligned} B &= M_2 \cdot N_1 + N_2, \quad 0 \leq N_2 < N_1 \\ N_1 &= M_3 \cdot N_2 + N_3, \quad 0 \leq N_3 < N_2 \end{aligned}$$

.....

$$N_{k-2} = M_k \cdot N_{k-1} + N_k, \quad N_k = 0$$

Тогава N_{k-1} е НОД на A и B . Докато първият алгоритъм очевидно намира НОД на две , Евклид се нуждае от доказателство.

Да илюстрираме работата на алгоритъма на Евклид с един пример.

Нека $A = 784$, $B = 240$. Последователно получаваме:

$$784 = 3 \cdot 240 + 64$$

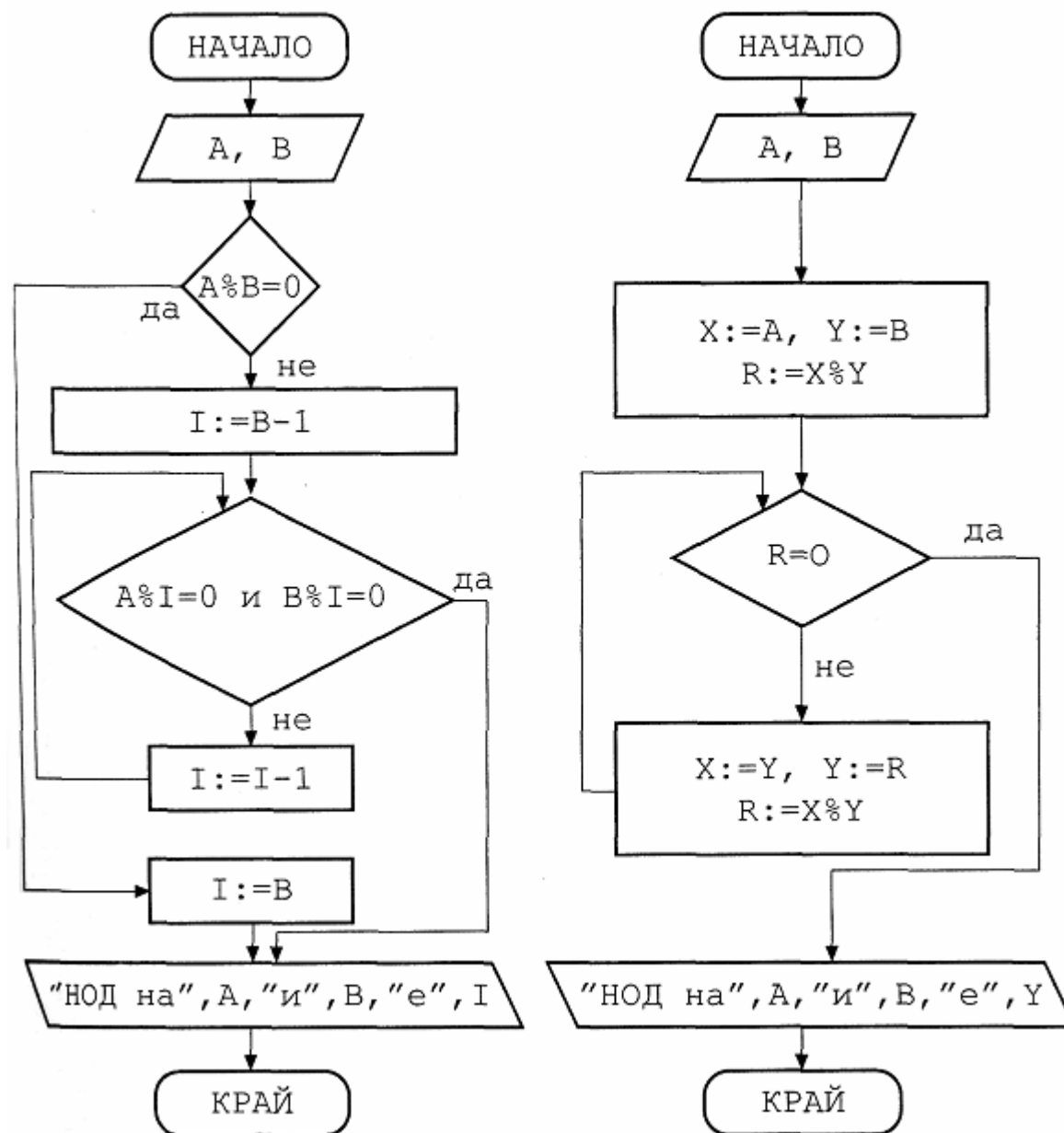
$$240 = 3 \cdot 64 + 48$$

$$64 = 1 \cdot 48 + 16$$

$$48 = 3 \cdot 16 + 0$$

Следователно НОД на 784 и 240 е 16 .

Със знак $\%$ в блоковите схеми е означена операцията пресмятане на остатък при делени



Блоковите схеми решават задачата за намиране на НОД.

За много масови задачи първият алгоритъм, за който се досетим, не е най-добрият, е свързано с изучаването на свойствата на входните или изходните обекти. В редица случаи въобще не е възможно създаването на алгоритъм.

[към тема 14](#)