

ЛЕКЦИЯ 5

МЕТОДИ ЗА ЗАПИС НА АЛГОРИТМИ

-  **Използване на естествен език**
-  **Видове заповеди в записа**
-  **Използване на блок-схема**
-  **Използване на алгоритмичен език**
-  **Сравняване на методите**

АЛГОРИТЪМ

Определение на А. А. Марков
(основоположник на съвременната
теория на алгоритмите):

**Алгоритъм се нарича точно и
общоразбираемо предписание,
определящо изпълнението
на последователност от елементарни
операции, чрез които се решава клас
от еднотипни задачи.**

ЕСТЕСТВЕН ЕЗИК

-  Изписваме **заповедите** за действие **на естествен (човешки) език**.
-  Можем да **използваме и** познатите **математически означения** (+, −, >, ≠).
-  Приемаме, че **написаното** ще може да бъде разбрано **от всеки, който говори** използвания **език** и е учил математика.
-  **Възможно е** да попаднем на **изпълнител**, който **не може** да извърши **всички** означени **действия** (напр. коренуване).
-  Възможно е **двусмислено тълкуване**.

ЕСТЕСТВЕН ЕЗИК (прод.)

-  За да можем лесно да указваме коя заповед е следваща, **при писането** им ги **номерираме с естествени числа**.
-  Изпълнението **започва от** заповед **①**.
-  Когато **след изпълнение** на една заповед трябва да се изпълни **тази, чийто номер е с единица по-голям** от нейния, това може да не се записва явно в самата заповед.
-  В една заповед **може да няма** указано **друго действие освен посочване** на номер **на следващата** за изпълнение.

АЛГОРИТЪМ НА ЕВКЛИД ①

Вход: Две естествени числа a и b .

Изход: НОД(a , b).

- ① Въведи двете числа a и b . Изпълни ②.
- ② Ако $a > b$, изпълни ⑤, иначе изпълни ③.
- ③ Ако $a = b$, изпълни ⑥, иначе изпълни ④.
- ④ Пресметни $b - a$ и в бъдеще считай, че това е стойността на b . Изпълни ②.
- ⑤ Пресметни $a - b$ и в бъдеще считай, че това е стойността на a . Изпълни ②.
- ⑥ Съобщи стойността на a . Изпълни ⑦.
- ⑦ Прекрати изпълнението на алгоритъма.

ВИДОВЕ ЗАПОВЕДИ

 Указание за **действие** и указване на  **следваща**, когато тя е **единствена** [**4** **5**].

 Указание за **действие**, когато **номерът**  **на** единствената **следваща** е **по-голям** **с единица** [**1** **6** *можеха да са*].

 **Само** указване на **следваща** [*няма*].

 Указание за **проверка на условие** и  указване на **две следващи** [**2** **3**].

 Указание за **завършване** (край) [**7**]. 

ЕЗИК НА БЛОК-СХЕМИТЕ

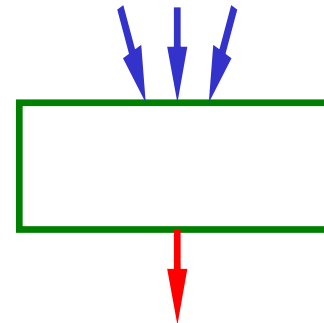
-  **Графичните изображения** се възприемат от хората по-леко.
-  **Структурата** на алгоритъма може да бъде проследена по-лесно.
-  В **различно оформени блокове** се записват словесно или с математически символи сведения за заповедите.
-  **Редът на изпълнение** се посочва **чрез стрелки**, които свързват двойката последователно изпълнявани блокове.

ОСНОВНИ БЛОКОВЕ

Начало на
изпълнението



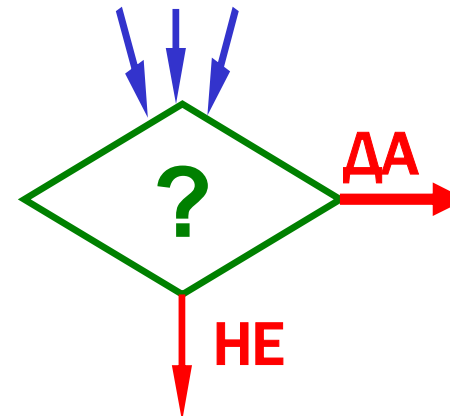
Извършване
на действия



Заповед за край
на изпълнението

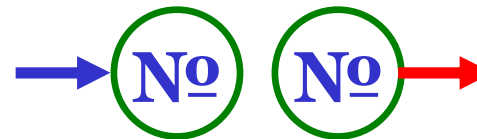


Проверка
на условие



СПОМАГАТЕЛНИ БЛОКОВЕ

Прекъсване
на стрелка



Подалгоритъм
(отделна схема)



Вход или
изход



Вход
(перфокарта)



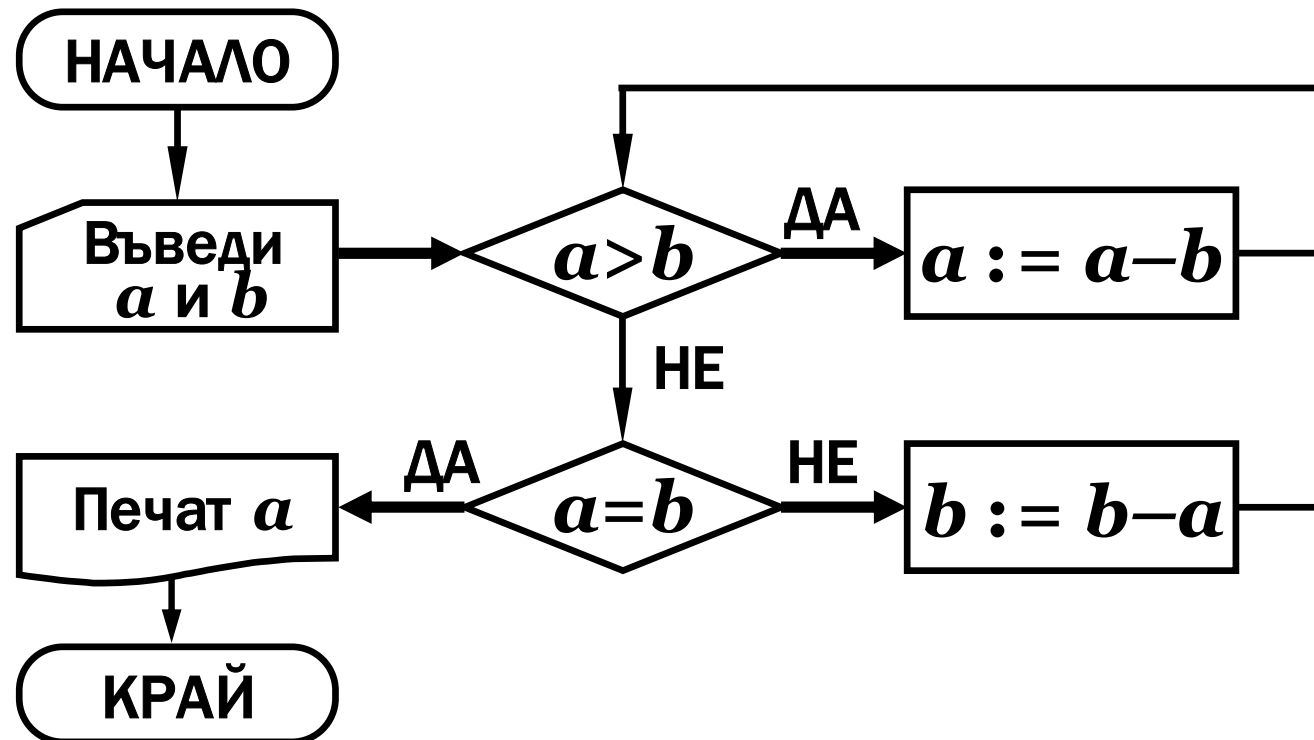
Изход (печат
на хартия)



АЛГОРИТЪМ НА ЕВКЛИД ②

Вход: Две естествени числа a и b .

Изход: НОД(a , b).



ПРОГ_05

10/18

АЛГОРИТМИЧЕН ЕЗИК

-  Създаваме **специализиран език за запис на алгоритми.**
-  Езикът е **линеен** (като човешките).
-  Освен действията с езика могат да се записват и **особеностите на данните.**
-  Достъпните **конструкции и смисълът им са определени** предварително.
-  Не се допуска **двусмислено** тълкуване.
-  Езикът може да отчита **особеностите на описваните с него алгоритми.**

АЛГОРИТЪМ НА ЕВКЛИД ③ (ПАСКАЛ)

```
VAR A, B: INTEGER;  
BEGIN  
    READLN (A, B) ;  
    WHILE A<>B DO  
        IF A>B THEN A := A - B  
        ELSE B := B - A;  
    WRITELN ( 'НОД е ' , A) ;  
END .
```

АЛГОРИТЪМ НА ЕВКЛИД ④ (СИ)

```
main()  
{  
  int a, b;  
  scanf(a,b) ;  
  while a<>b  
    if (a>b) a = a - b;  
    else b = b - a;  
  printf("НОД е ", a) ;  
}
```

АЛГОРИТЪМ НА ЕВКЛИД ⑤ (ВИЖУЪЛ БЕЙСИК)

```
Dim A, B As Integer
Input #1, A, B
Do While A <> B
    If A > B Then
        Let A = A - B
    Else
        Let B = B - A
    EndIf
Loop
Print #2, "НОД е "; A;
```

ПРЕДИМСТВА И НЕДОСТАТЪЦИ НА ЕСТЕСТВЕНИЯ ЕЗИК

- 😊 **Не** е необходимо да **се изучава**.
- 😞 **Не** е разбираем **за машина**.
- 😞 Словесните изрази **не са строго и еднозначно определени**.
- 😞 Има възможности за **двусмислено тълкуване**.
- 😞 **Структурата** на алгоритъма **не се вижда ясно**.

ПРЕДИМСТВА И НЕДОСТАТЪЦИ НА БЛОК-СХЕМИТЕ

- 😊 **Нагледни и обозрими** (до **1 лист!**).
- 😊 **Ясна структура** на алгоритъма.
- 😊 Изпълнението се **следи лесно**.
- 😞 **Действията** се описват **словесно**.
- 😞 **Не са** пригодени **за машина**.
- 😞 **Лисват** блокове за **деклариране** на входните **данни и резултата**.
- 😞 **Преплитането** на линиите **влошава** **структурата** и **разчитането**.

ПРЕДИМСТВА И НЕДОСТАТЪЦИ НА АЛГОРИТМИЧНИТЕ ЕЗИЦИ

- 😊 Ясно, точно и еднозначно описание.
- 😊 Възможно е описание на данните.
- 😊 Пригодни за четене от машина.
- 😊 Изразните средства могат да бъдат съобразени с даден клас алгоритми.
- 😞 Трябва да се учат допълнително.
- 😞 Провокират допускане на грешки.
- 😞 Не винаги са удобни за хората.

**БЛАГОДАРЯ ВИ
ЗА ВНИМАНИЕТО!**

**БЪДЕТЕ С МЕН И
В СЛЕДВАЩАТА ЛЕКЦИЯ,
КОЯТО ЩЕ НИ ОТВЕДЕ
В НЕВЕРОЯТНИЯ СВЯТ НА
КЛАСИФИКАЦИЯТА
НА ЕЗИЦИТЕ**