

Алгоритми

УРОК ПО ИНФОРМАТИКА ЗА 9 КЛАС

изготвил: инж. В. Митева

Съдържание

- Същност на алгоритмите
- Свойства на алгоритмите
- Видове алгоритми
- Начини за описание на алгоритмите
- Въпроси
- Задачи за самоподготовка
- Използвана литература

1. Същност на алгоритмите

Всеки от нас знае как се мият ръце:

- пуска се водата;
- насапунисват се ръцете под течаща топла вода;
- търкат се ръцете една в друга;
- изплакват се обилно;
- спира се водата;
- изсушават се ръцете.



 *Запомни!*

Последователността от инструкции, чието изпълнение води до определен резултат, след краен брой стъпки, се нарича **АЛГОРИТЪМ**.

Исторически сведения

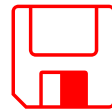
- ✓ Терминът **алгоритъм** произлиза от латинизираната форма на името на арабския математик **Абу Джафар Мохамед ибн Муса ал-Хорезми**, който около 820 г. от н. е. написва научен трактат за това, как да се представят (записват) числата в десетичната бройна система и как да се смята с тях.
- ✓ Произведението на Ал Хорезми (al-Gorezmi) е преведено на латински език в Средновековна Европа в началото на XII век, но разпространението на „изкуството да се смята“ и ерата на ръчното и механизирано смятане започва едва през XVI век.

ОСНОВНИ ПОНЯТИЯ



Запомни!

Всяка инструкция в алгоритъма се нарича **елементарно действие**, което може да се извършва без допълнителни указания.

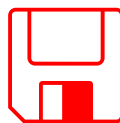


Запомни!

Всяко изпълнение на елементарно действие, се нарича **стъпка**.

ПОДАЛГОРИТМИ

- При по-сложните алгоритми се налага една и съща последователност от действия да се описва и изпълнява няколко пъти на различни места.
- Избягването на това повторение става чрез обособяване на тази последователност от действия като самостоятелен алгоритъм.



Запомни!

Подалгоритъм е самостоятелно обособена поредица от елементарни действия, която може да се използва в общия алгоритъм многократно.

2. Свойства на алгоритмите

- 2.1. **Дискретност** – всеки алгоритъм се състои от краен брой елементарни действия.
- 2.2. **Яснота** - всяка текуща стъпка трябва да е точно указана и еднозначно да бъде определена следващата.
- 2.3. **Формалност** – изпълнителят може да работи формално, да изпълнява указанията докато стигне до указание за край.
- 2.4. **Определеност** – при едни и същи данни се получава един и същи резултат.

2.5. Масовост – всеки алгоритъм да се прилага успешно при решаване на задачи с различни входни данни.

2.6. Крайност – всеки алгоритъм да завършва с резултат за краен брой стъпки.

2.7. Резултатност – всеки алгоритъм трябва винаги да дава резултат, ако входните данни принадлежат на допустимото подмножество.

3. Видове алгоритми

3.1. Линејни



Запомни!

Алгоритми, в които действията се изпълняват последователно по реда, в който са записани, се наричат

ЛИНЕЙНИ

ПРИМЕР:

Да се определи обиколката на триъгълник със страни a , b и c .

Входни данни: a , b , c , $a > 0$, $b > 0$, $c > 0$;

Резултат: P – обиколка;

Стъпка 1: задава се стойност на a ;

Стъпка 2: задава се стойност на b ;

Стъпка 3: задава се стойност на c ;

Стъпка 4: $P = a + b + c$.

3.2. Разклонени



Запомни!

Алгоритми, в чието описание има наличие на условие за проверка се наричат

РАЗКЛОНЕНИ

ПРИМЕР:

Да се разделят две числа a и b .

Входни данни: a и b ;

Резултат: C ;

Стъпка 1:

Задава се стойност на a .

Стъпка 2:

Задава се стойност на b ;

Стъпка 3:

Ако $b=0$, делението е невъзможно,
в противен случай: $C=a/b$

3.3. Циклични



Запомни!

*Алгоритми, при
чието изпълнение се
повтарят
няколкократно част
от инструкциите, се
наричат*

ЦИКЛИЧНИ

ПРИМЕР:

Да се намери сумата на числата от 1 до 100.

Входни данни: целите числа от 1 до 100 и br ;

Резултат: S - сумата ;

Стъпка 1: $S=0$

Стъпка 2: $br=1$

Стъпка 3: $S=S+br$

Стъпка 4: Ако стойността на брояча е по-малка или равна на 100, то тя да се увеличава с 1 и се преминава към стъпка 3.

В противен случай – край.

4. Начини за описание на алгоритмите

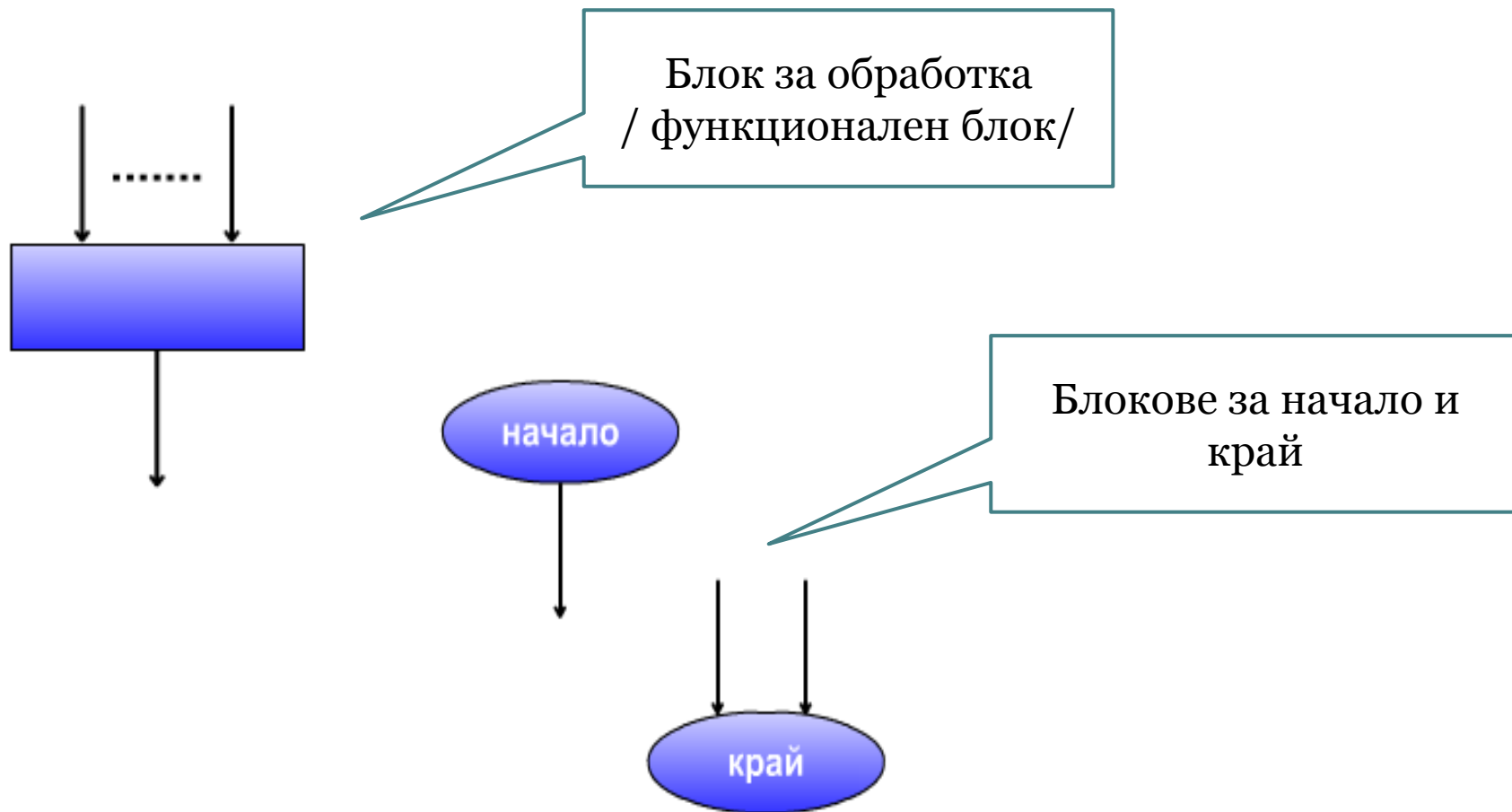
- ✓ **Словесно** – на естествен език се описват една по една стъпките.
- ✓ **Графично** – с помощта на блок-схеми
- ✓ **Посредством език за програмиране** – Паскал, C++ и т.н.

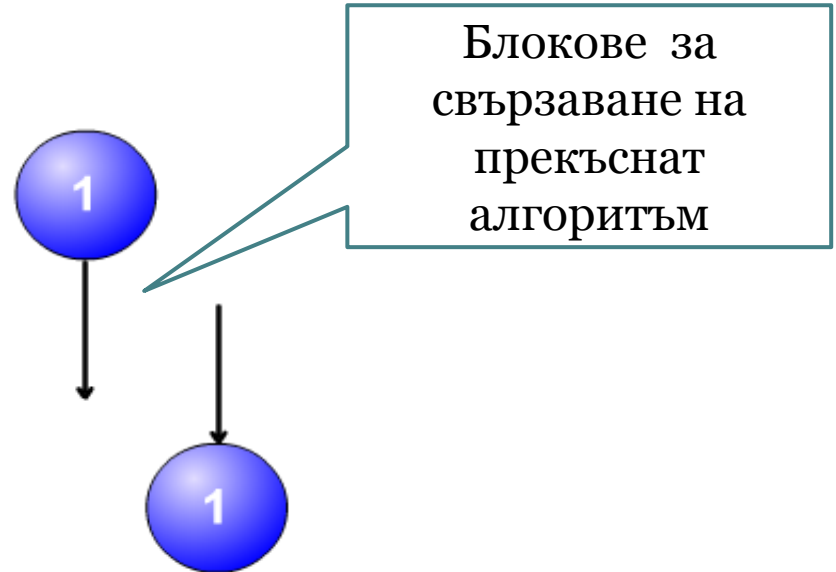
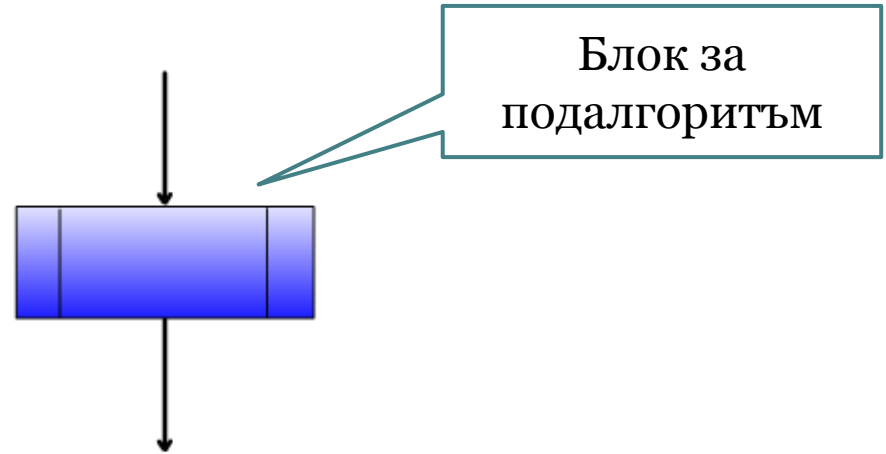
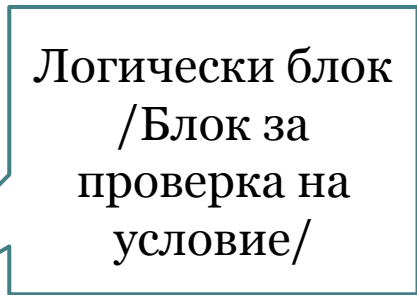
5. Блок-схеми

 *Запомни!*

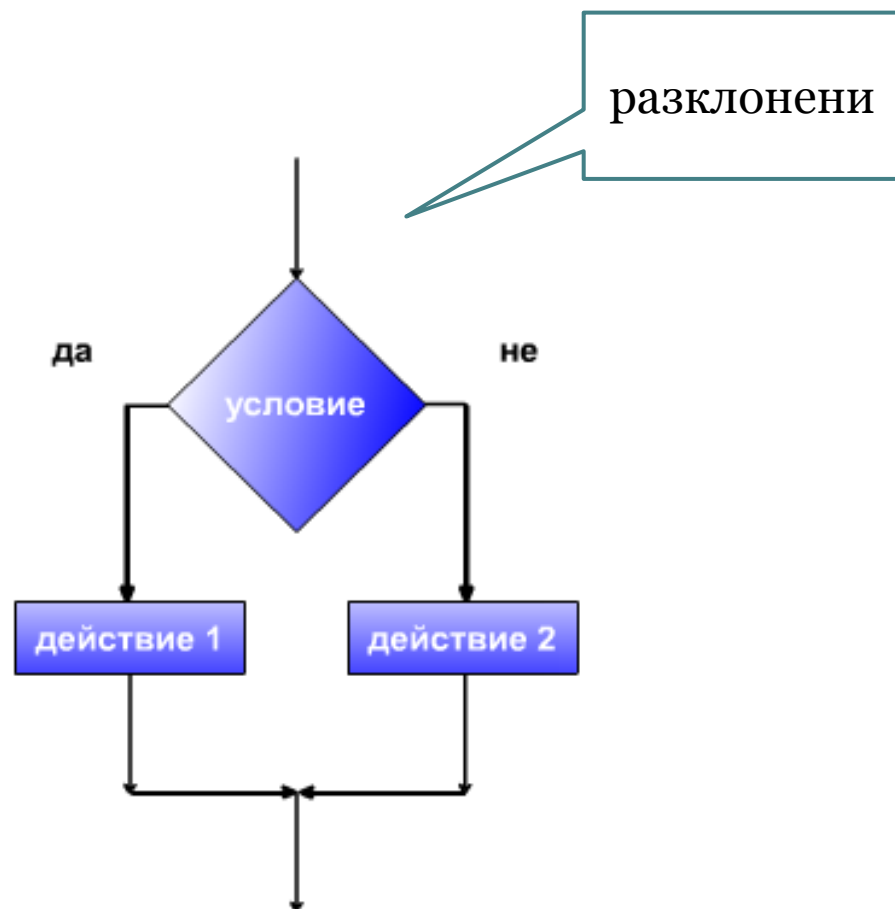
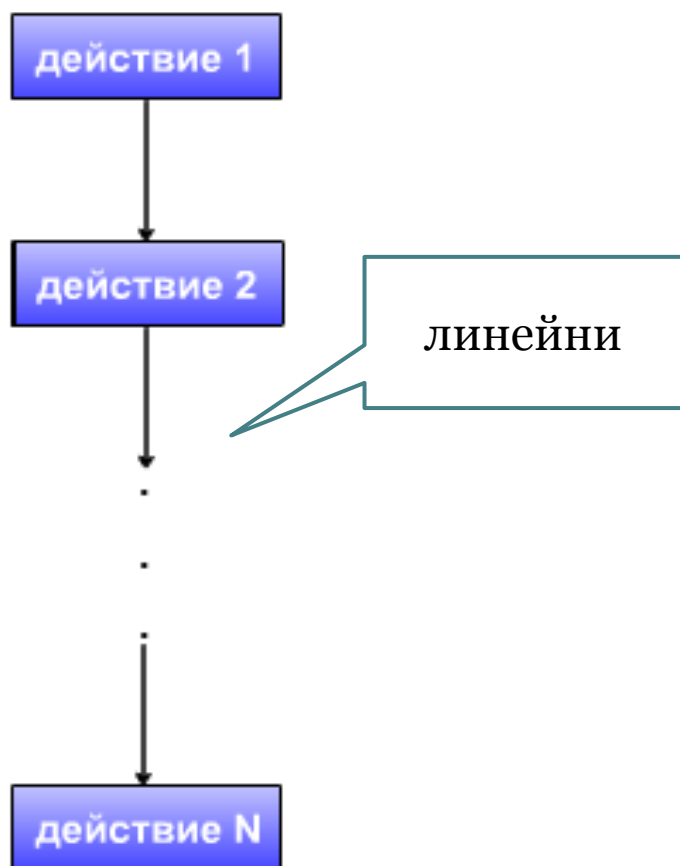
Блок-схемите се състоят от блокове и свързващи ги стрелки. В блоковете се описват действията на алгоритъма, а свързващите ги стрелки указват последователността на изпълнение на действията.

5.1. Видове блокове





5.2. Базови алгоритмични конструкции



Циклични



Цикъл с постусловие



Примерна задача:

Дадена е окръжност с радиус R . Да се намери лицето на окръжността с радиус R , в противен случай да се изведе подходящо съобщение

Словесно описание

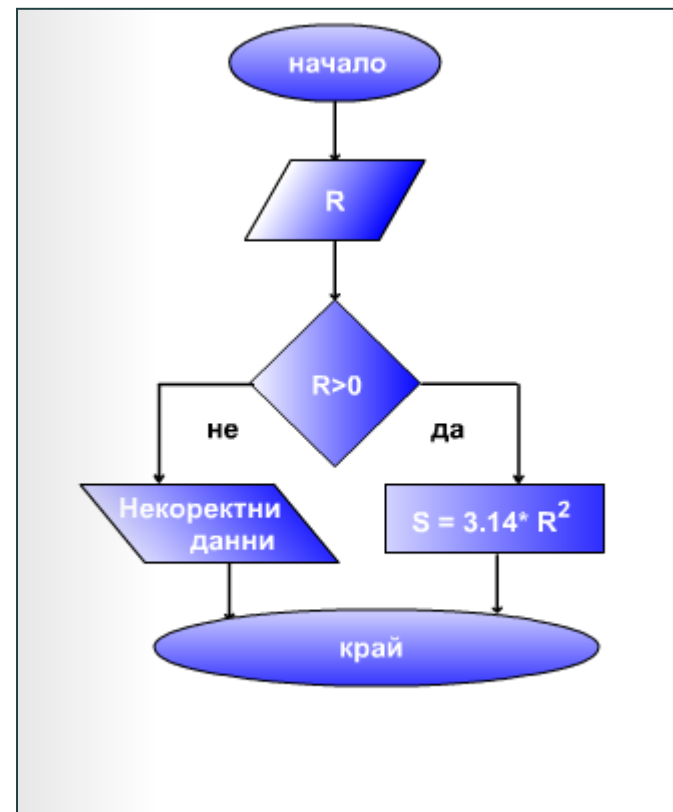
Стъпка 1: Въвеждаме R

Стъпка 2: Проверяваме знака на R :

- ако $R > 0$, изчисляваме $S = 3.14 * R * R$
- в противен случай извеждаме съобщение за некоректно въведени данни

Стъпка 3: Извеждаме S

Графично описание чрез блок-схема



Алгоритъм на Евклид

Алгоритъмът намира най-големият общ делител / НОД/ на две числа a и b

Словесно описание:

1. Ако $a > b$, то изпълни стъпка 2, в противен случай – стъпка 3.
2. На a дай стойността на разликата на a минус b . Изпълни стъпка 1.
3. Ако $b > a$, то изпълни стъпка 4, в противен случай – стъпка 5.
4. На b дай стойността на разликата на b минус a . Изпълни стъпка 1.
5. Най-големият общ делител е $a=b$.

Задача1. Изпълнете алгоритъма за числата 6 и 15.

Задача2. Изразете алгоритъма чрез блок-схема.

Въпроси

- ✓ Какво означава понятието “АЛГОРИТЪМ”?
- ✓ Кои са основните свойства на алгоритмите?
- ✓ Какви са видовете алгоритми?
- ✓ Кои са начините за описание на алгоритмите?
- ✓ Какво представляват блок-схемите?
- ✓ Кои са основните блокове в тях?

Задачи за самоподготовка

Задача 1.

Да се състави алгоритъм за решаване на линейно уравнение

$$ax + b = 0$$

Задача 2.

Да се състави алгоритъм за пресмятане на функцията y :

$$y = -13x - 3, \text{ за } x < -3$$

$$y = -11x + 3, \text{ за } x \geq -3$$

Задача 3.

Да се състави алгоритъм, който проверява дали дадени три отсечки a , b и c могат да са страни на триъгълник и ако “да”, то да се извежда видът му.

Задача 4.

Да се състави алгоритъм, пресмяташ сумата на редицата

$$1x + 2x^2 + 3x^3 + \dots + nx^n$$

Използвана литература:



Съдържание