

Программалауға кіріспе. Алгоритмілер және программалауға кіріспе.

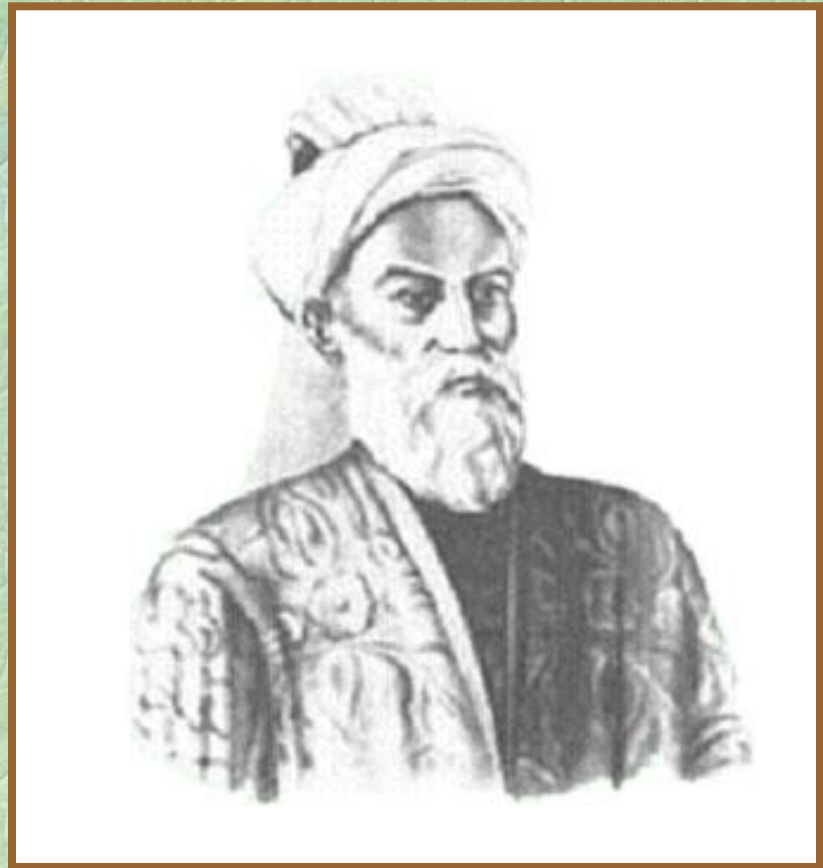
Баймаханова А.Б

Мақсаты

- *Алгоритм ұғымы, қасиеттері.*
- *Алгоритмдердің типтері, берілу ерекшеліктері.*
- *Алгоритмдерді құрудың негізгі тәсілдері.*
- *Бағдарламалау тілдері, олардың жіктелуі.*

Алгоритм ұғымы информатика пәнінің іргелі ұғымдарының бірі болып есептеледі.

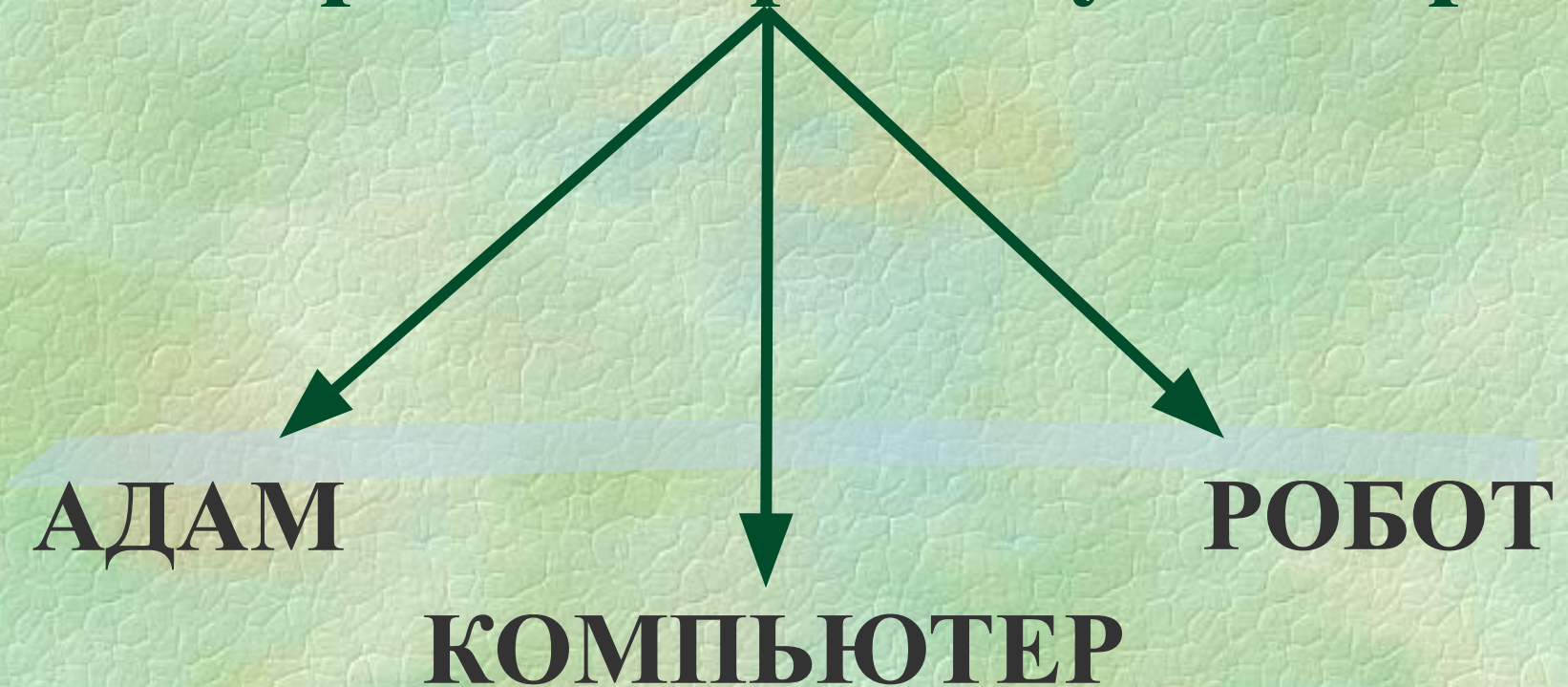
«Алгоритм» ұғымы IX ғасырда (783 – 850 ж) ондық санау жүйесіндегі арифметикалық амалдарды орындау ережесін (алгоритмін) алғаш жазған, Хорезм елінің тумасы Мұхаммед Мұса-ұлы атты араб математигінің есіміне байланысты пайда болған. Ол - өзінің “Арифметикалық трактат” деген еңбегінде арифметикалық амалдарды орындау тәртібін ұсынған.



Анықтама:

*Алгоритм – орындаушыға
ұғынықты тілмен, қандайда бір
берілген есепті шешуге
болатындай етіп жазылған
нақты бұйрықтар тізімі*

Алгоритмді орындаушылар



Орындаушы алгоритмді формальді түрде орындайды

Алгоритм қасиеттері:

- *Анықтылығы*
- *Жаппайлығы*
- *Нәтижелілігі*
- *Дискреттілігі*
- *Түсініктілігі*

Анықтылығы — алгоритмнің нұсқаулары және олардың орындалу реті бір мағыналық, бір мәндік түсінік беруге тиіс. Орындалу үрдісінде алгоритм еркіндікке жол бермеуі керек.

Жаппайлығы (бірдейлігі) – типі берілген қандай да бір есептер тобын шешуге қолдану мүмкіндігі қарастырылған және берілген аралықта бастапқы деректердің мәндері өзгергенде алгоритмнің дұрыс жұмыс жасауы.

Нәтижелілігі (шектілігі)– саны шектеулі реттелген нұсқауларды орындап аяқтағанда тиынақты бір нәтижеге жеткізуге тиіс.

Дискреттілігі – қарастырылып отырған ақпаратты өңдеу үрдісі ретімен жазылған тізбекті жеке-жеке қадамдарға бөлінуге тиіс. Әр қадам ол анық бірінен-бірі бөлек нұсқау (команда).

Түсініктілігі – орындаушы (адам, ЭЕМ) нұсқаулар жүйесі арқылы жазылған алгоритмді түсініп, орындай алатындығы.

Алгоритмдердің негізгі түрлері

СЫЗЫҚТЫҚ

тармақталған

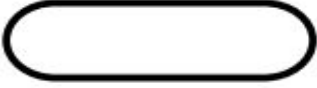

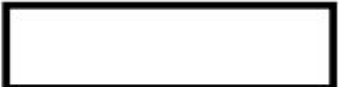

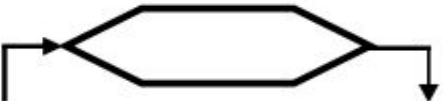

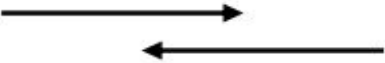
циклдік

Мұнда бұйрықтар бірінен соң бірі ілесу тәртібімен орындалады

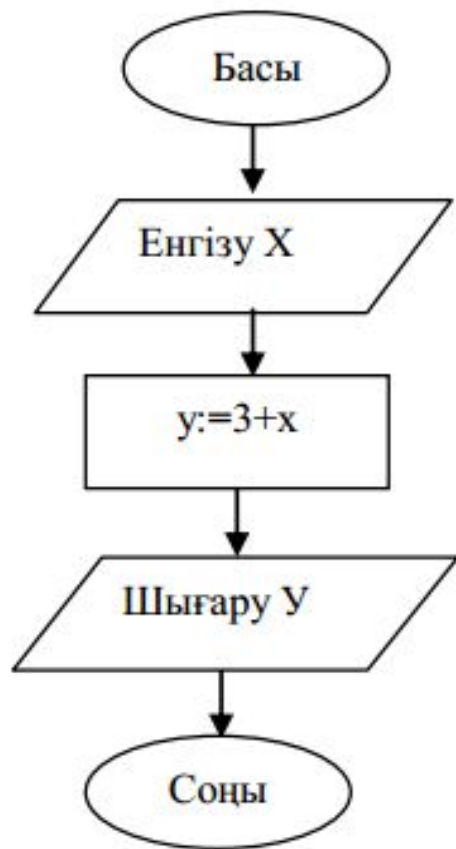
Мұнда есепті шығару барысында кейбір шарттарды таңдау мүмкіндігі болады

Жеке бұйрықтар немесе бұйрықтар тобы бірнеше рет қайталанады.

1-кесте. Блок-схемадағы шартты белгілеулер

Геометриялық пішім	Оның атқаратын жұмысы
	Алгоритмдерді бастау және аяқтау
	Мәліметтерді енгізу мен нәтижені шығару
	Математикалық өрнектерді есептеу
	Шартты тексерудің нәтижесінде есептеу процесі бағыттың бірі бойынша жалғастырылады
	Қайталануды ұйымдастыру
	Қосалқы бағдарламаларға кіру және шығу
	Блоктарды бір-біріне байланыстырады

СЫЗЫҚТЫҚ алгоритм түрі



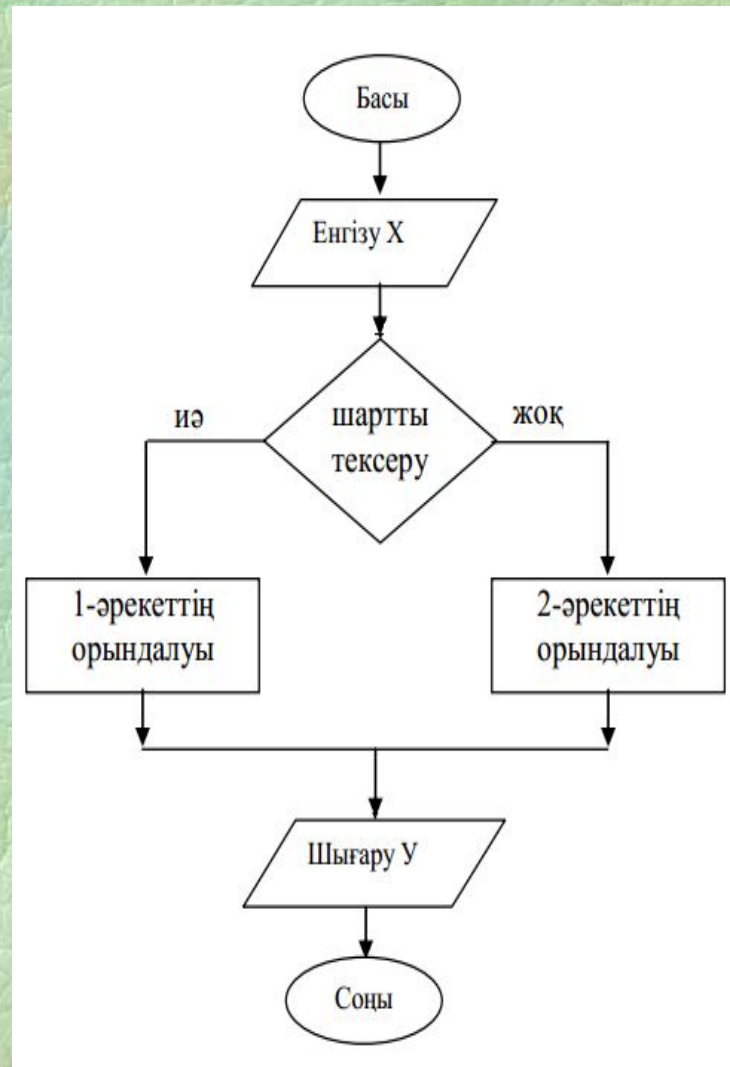
Блоктары ретімен орындалатын алгоритм сызықтық алгоритм деп аталады немесе денесі тек ретімен орындалатын командалардан тұратын алгоритмді **СЫЗЫҚТЫҚ алгоритм** деп аталады.

Тармақталған алгоритм түрі

Берілген шарттың орындалуына байланысты бір тармақ бойынша орындалатын есептеу процестері бар алгоритмдер тармақталған алгоритм деп аталады.

Тармақталған алгоритмдерде тармақталу және таңдау командалары қолданылады.

Тармақталу командасы шарттың орындалуына байланысты екі әрекеттердің бірін орындауға нұсқау береді.

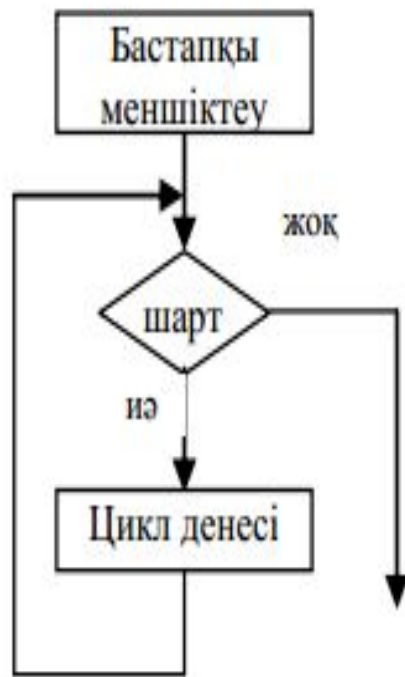


Циклдік алгоритм түрі

- Цикл денесі деп аталатын әрекеттер тізбегін көп рет қайталануын іске асырған алгоритмді **циклдік** деп атайды. Циклдер қайталану саны белгілі және алдын ала белгісіз деп бөлінеді. Кейде циклді неше рет қайталу керектігі алдын ала белгісіз, бірақ, қандайда бір шарт әзірше әділ болса, оның орындалатыны белгілі. Қайталау саны алдын ала белгілі және белгісіз циклдік құрылымды алгоритмдердің блок-схемалары бейнеленген.
- **Әзірше циклы** – цикл денесі орындалғанға дейін шарт тексеріледі, егер бірінші тексерісте циклдан шығу шарты орындалмаса, онда цикл денесі бір ретте орындалмайды.
- **Дейін циклы** – қандайда бір шарт орындалғанға дейін, қандай да болса есептеулерді бірнеше рет орындау қажеттігінде қолданылады. Бұл циклдың ерекшелігі, цикл денесі ылғи ең жоқ дегенде бір рет орындалады.
- **Параметрлі цикл** – бүтін санды параметр бастапқы (In) мәнінен соңғы (Ik) мәніне дейін белгілі қадаммен өткенге дейін цикл денесі орындалады.

Циклдік құрылымды алгоритмдердің блок-схемалары

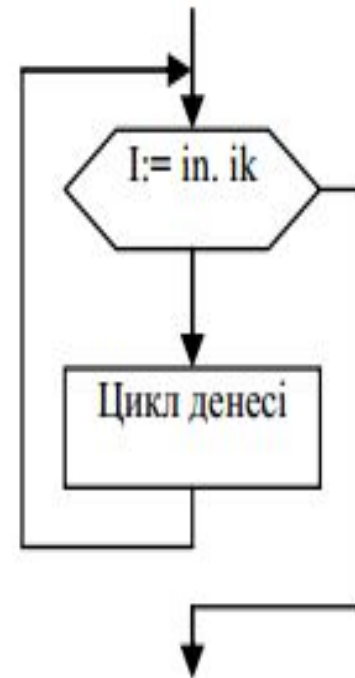
Әзірше циклы



Дейін циклы



Параметрлі цикл



Берілу тәсілдері

```
graph TD; A[Берілу тәсілдері] --> B[Сөздік тәсіл]; A --> C[Алгоритмдік тіл немесе бағдарлама]; A --> D[Блок-схема];
```

Сөздік тәсіл

Алгоритмдік тіл
немесе бағдарлама

Блок-схема

Бағдарламалау тілі

Бағдарламалау тілі дегеніміз адамның ЭЕМ-мен байланысының жасанды тілі, ол мәліметтер мен оларды өңдеу алгоритмдерін компьютерде сипаттауға арналады.

Егер бағдарламалау тілі нақтылы процессор типіне арналған және оның ерекшеліктерін ескеретін болса, онда ол *төмен деңгейлі бағдарламалау тіліне* жатады.

Жоғары деңгейлі бағдарламалау тілі компьютерге қарағанда, адамға анағұрлым жақын әрі түсінікті болып келеді.

Жасанды тілді компьютер өзінің машиналық тіліне адамның қатысуынсыз автоматты түрде аударатындай етіп жасайды. Аударуды программа орындайды.

Ондай программаны ағылшын тілінде **«транслятор»** (аудару) деп атайды, ал аудару процесін «трансляция» дейді.

Компиляция әдісі – алдымен жасанды программа тнгелдей машиналық тілге аударылады, сонан кейін бұл программа басынан орындалады.

Интерпретация әдісі – жасанды тілдегі программаның әрбір әрекеті (нұсқау немесе амалы) жеке алдын-ала аударылмастан бірден машиналық тілде орындала бастайды. Яғни, компиляция әдісінде жасанды тілдегі программа тек қана бір рет қарастырылады және аударылған программаны бірнеше рет орындауға болады, ал интерпретация әдісінде әр орындар алдында жасанды тілдегі программаны қайтадан қарастыру қажет.

Матиматикалық функциялар

Функция аты	Математикалық	Бейсик тілінде	Бейсик тілінде
Абсолюттік шама	$ x $	ABS(X)	ABS(-5)
Экспонента	e^x	EXP(X)	EXP(5)
Натуралдық логарифм	$\ln x$	LOG(X)	LOG(3)
Қалдық (MOD)	-	$x \text{ MOD } y$	19 MOD 6.7
Квадраттық түбір		SQR(X)	SQR(4)
Синус	$\sin x$	SIN(X)	SIN(3.14)
Косинус	$\cos x$	COS(X)	COS(1)
Тангенс	$\text{tg } x$	TAN(X)	TAN(2)
Арктангенс	$\text{arctg } x$	ATN(X)	ATN(3)
Сандық белгілер	-	SGN(X)	SGN(-5)

Квадрат теңдеуді шешу алгоритмі мен блок-схемасы және бағдарламасын жазу

$$ax^2+bx+c=0$$

a, b, c -кез келген сан ($a \neq 0$)

$D < 0$ - шешімі жоқ

$D = 0$ -бір шешімі бар

$D > 0$ -екі шешімі бар

$$D = b^2 - 4ac$$

$$x_1 = \frac{-b + \sqrt{D}}{2a}$$

$$x_2 = \frac{-b - \sqrt{D}}{2a}$$

Алгоритмі

алг (арг нақ а, в, с, нәт нақ x_1, x_2, y)

басы нақ D

$$D := b^2 - 4ac$$

егер $D < 0$

онда $y :=$ “шешімі жоқ”

әйтпесе егер $D = 0$

онда $x := -\frac{b}{2a}$

әйтпесе

$$x_1 := \frac{-b + \sqrt{D}}{2a}$$

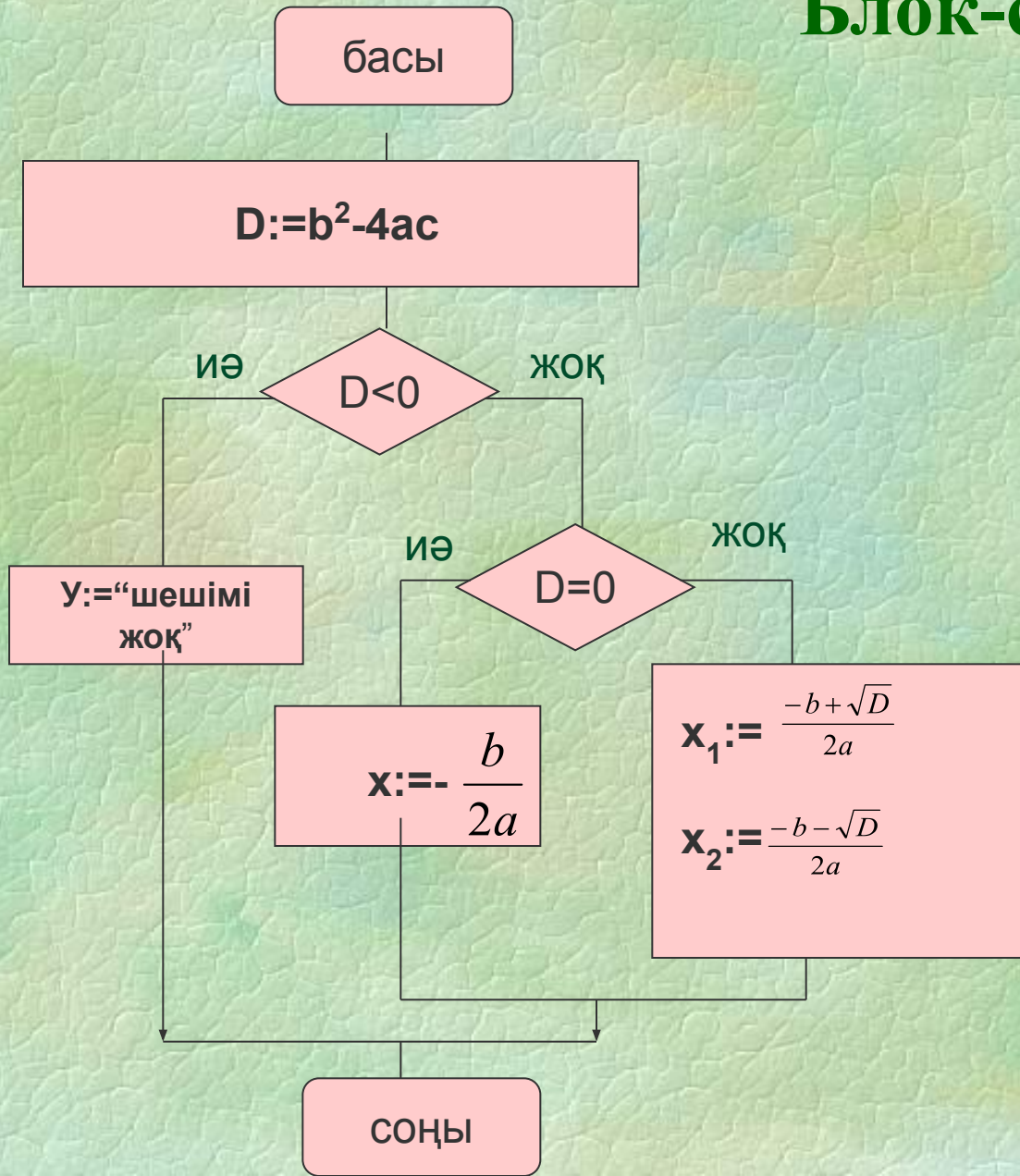
$$x_2 := \frac{-b - \sqrt{D}}{2a}$$

бітті

бітті

соңы

Блок-схема



Программалау тілі

```
Program esep 1;  
var a,b,c,d:real;  
begin  
    writeln('a,b,c engis');  
    readln(a,b,c);  
D:=sqr(b)-4*a*c  
If d<0 then writeln('жок')  
    else begin writeln('бар');  
                writeln('x1:=(-b+sqrt(d))/(2*a)');  
                writeln('x2:=(-b-sqrt(d))/(2*a)');  
    end;  
end.
```


Программалау тілдерінің бір – бірімен байланысу схемасы

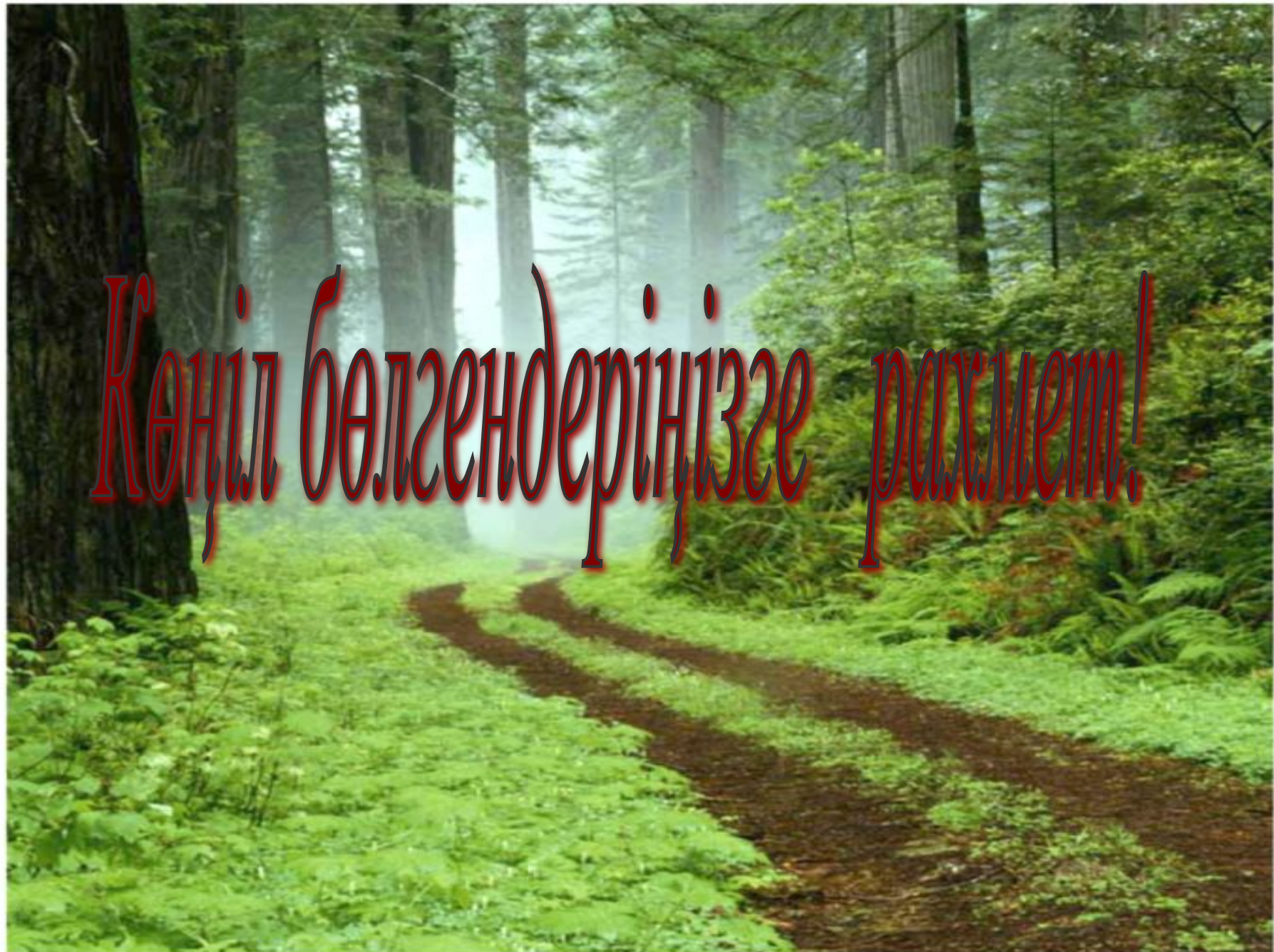


Қазіргі кезде ең көп тараған негізгі программалау тілдеріне қысқаша тоқталып өтейік.

- **Ада** – 1980 ж. шықты, жүйелік және іс жүзінде программа жасауға болатын әмбебап тіл.
- **Алгол** – ағылшын әмбебап программалу тілдерінің бірі.
- **Ассемблер** – машинаға бейімделген программалау тілі.
- **Бейсик** – есепті ЭЕМ – де шығару үшін, программаны “диалог” режимінде құрады.
- **Кобол** – экономикалық есептерді шығаруға өте қажетті программалау тілі болып дүние жүзіне көп тараған тіл.

Қазіргі кезде ең көп тараған негізгі программалау тілдеріне қысқаша тоқталып өтейік.

- **Паскаль** – блокты, модулды программа құрамын құрастыруға қажетті тіл. Алгол тілінің әр қарай даму жолындағы жалғасы болып саналады.
- **PL – 1** - әмбебап, процедуралы бейімделген тіл, әр түрлі информацияны өңдеуге, әр түрлі есептерді шығаруға үлкен мүмкіншіліктері бар өте ықтималды тіл.
- **Си** – машинаға тәуелсіз тіл. Жүйелік программалар құру үшін және программамен қамтамасыз ету жүйесін бір машинадан екінші машинаға көшіруге, жазып алуға қажетті тіл.
- **Фортран** – бірінші жасалған тіл. Қазіргі кездегі жоғарғы сатыдағы есептегіш программаларын құруда көп таралған тілдің бірі.



Көңіл бөлгендеріңізге рахмет!