

## **C++ тилинин негизги түшүнүктөрү жана программа түзүү.**

C++ тилинин ариптери. Ариптер катары ASCII коддоо таблицасынын символдорунун тобу кирет.

**Ысымдар.** C++ тилинде ысымдар менен өзгөрмөлөрдү, функцияларды, белгилерди белгилешет. Ысымдар төмөндөгү эрежелерди эске алуу менен түзүлөт:

- Ысымдар латын ариптеринен, цифралардан жана астынкы сызык «\_» белгисинен түзүлөт.

- Библиотекалык функцияларда, системдик өзгөрмөлөрдө жана турактууларды белгилөөдө биринчи белгиси «\_» астынкы сызык менен башталгандыктан, ысымдарды бул белги менен белгилөөдөн качуу керек.

- C++ тилинде латын тамгаларынын баш тамгалары жана кичине тамгалары айырмалангандыктан name жана NAME ысымдары ар башкача болот.

**Программалоо** – компьютер программаларын түзүү процесси. Тар мааниде программалоо конкреттүү программалоо тилинде (көбүнчө мурда болгон алгоритм – коюлган милдетти чечүү планы, методу боюнча) программаларды жазуу.

### **MS Visual C++ чөйрөсү**

Белгилүү болгондой, программалоо тилин үйрөнүүнүн эң жакшы жолу бул тилге программаларды жазуу жана алардын компьютерде иштөөсүн текшерүү. Бул бир нече программаларды талап кылат:

1. **Тексттик редактор**, C++ программасынын баштапкы кодун терип жана анын жардамы менен түзүлөт.
2. **Компилятор**. Бул программа баштапкы текстти компьютер түздөн-түз аткара турган машина көрсөтмөлөрүнө айлантат.
3. **Шилтемечи**. Программанын айрым түзүлгөн бөлүктөрүн бир бүтүнгө чогултуучу жана зарыл болгон учурда аларга даяр китепканалардан компоненттерди кошо турган шилтеме берүүчү. Натыйжада, макет ишке даяр программа - аткарылуучу файл.
4. **Мүчүлүштүктөрдү ондоочу**, программадагы каталарды оңой табуу. Каталарды компиляция учурунда да, программаны аткаруу учурунда да табууга болот.

Visual C++ программаларды даярдалгандай түзөт жана иштетет.

**Долбоор**-программанын баштапкы файлдарын компиляциялоо жана аны стандарттык китепканалар менен байланыштыруу үчүн зарыл болгон бардык маалыматты камтыган маалымат структурасы.

### **Жаңы долбоор түзүү**

VisualC++ чөйрөсүндө жаңы долбоор түзүү үчүн:

1. File менюсунан NEW (File->New) атрибуттун тандаганда диалогдук терезеч чыгат.
2. Projects кыстырмасынын аткарылуучу файлдардын ар кандай түрлөрүнүн тизмесинен Win32Consoleapplication (Win32 платформалары үчүн консолдук тиркеме) тандалат.
3. Createnewworkspace баскычы белгиленгенин текшерип, андан кийин долбоордун атын (мисалы, тест), долбоордун аты талаасында (долбоордун аталышы) жана бул долбоорго тиешелүү бардык файлдар боло турган каталогдун аты тандалат.
4. Андан кийин, C++ программасынын тексти менен жаңы текст файлын түзүп, аны долбоорго кошулуп жана File менюсунан жаңы атрибут тандалат
5. Files кыстырмасынан C++SourceFile (исходный файл C++) жаңы файлдын тиби тандалат.

6. Файлдын аты сабында жаңы файлдын атын терилет (сpp кеңейтүүсү C++ баштапкы файлдары үчүн стандарт болуп саналат).  
Компиляция Build менюсү аркылуу ишке ашырылат. Buildtest.exe пунктун (же F7 функциялык баскычын) тандаңыз. Компиляция билдирүүлөрү экрандын ылдый жагында пайда болот. Компилятордун терезеси катаны көрсөтөт. Даяр программаны Build менюсун, Executetext.exe пунктун (же ctrl+F5) колдонууга болот.

### **C++ тилинде программанын түзүлүшү.**

C++ тилинде программа функциялардын тобунан турат жана программа негизги функциялардан башталат. Негизги функциянын ысымы main кыргызча которгондо башкы, негизги дегенди түшүндүрөт. Программаны аткаруу негизги функциялардан башталат. C++ тилиндеги программада main ысымдуу бир гана функция болушу керек.

C++ тилиндеги программанын жөнөкөй структурасы  
(түзүлүшү)

/*түшүндүрмө*/	1-сап
#include <iostream>	2-сап
main ( )	3-сап
{	4-сап
<ички баяндоолор>;	5-сап
<иш аракеттер>;	6-сап
}	7-сап

1-сапта /\*түшүндүрмө\*/ - C++ тилинде түшүндүрмөлөр программанын окулушун жана көрсөтмөлүүлүгүн жогорулатуу үчүн колдонулат. Түшүндүрмөлөр программанын каалаган жерине жазыла берет. Түшүндүрмөлөр /\*жана \*/ белгилеринин ортосунда жазылат.

2-сапта алгачкы процессордун буйругу б.а. программа компиляция болоор алдындагы иш аракеттерди жүргүзөт.

3-сапта негизги функциянын ысымы, ысымдан кийин тегерек каша сөзсүз коюлушу керек.

4-сапта «{» ачылган фигуралык кашаа негизги функциянын тулкусунун башталышы. Паскаль тилиндеги биринчи begin менен дал келишет.

5-сапта <ички баяндоолор> программада колдонулган өзгөрмөлөрдөн типтерин, көрсөткүчтөр ж.б. баяндайт.

6-сапта <иш аракеттер> операторлор, функциялар жазылат.

7-сапта «}» жабылган фигуралык кашаа негизги функциянын аякталышын билгизет.

Мисалы:

/* Менин C++ тилиндеги биринчи программам */	1-сап
# include <iostream>	2-сап
main ( )	3-сап

{	4-сап
std::cout<<"Саламатсызбы! \n";	5-сап
}	6-сап

Жогорку программада 1-сабы түшүндүрмөнү, 2-сабы алгачкы процессордун директивасын, 3-сабы негизги функцияны, 4-сабы негизги функциянын тулкусунун башталышын, 5-сап чыгаруу функциясын, ал эми 6-сабы негизги функциянын аякталышын кармат.

Бул программанын жыйынтыгы Саламатсызбы! деген сөздү экранга чыгарат жана жылгыч (курсор) жаңы саптын башына келтирилет.

*1-сап:* Түшүндүрмө берүүчү сап.

*2-сап:* Алгачкы процессордун буйруктарын берет.

**Алгачкы процессор**-программа компиляция болоор алдында, бир канча C++-стемдик буйруктарды аткаруучу C++-стемалык программа.

**Компиляция** - бул транслятордун бир түрү болгон компилятордун иштөө процесC++.

Программист жазган программа **бааштапкы модуль** деп аталат.

Машиналык буйрукка айланган программа **объектик модуль** деп аталат.

Кураштырылгандан кийинки программа б.а. курагыч(компаковка) иштегенден кийинки пайда болгон программа .exe же .com кеңейишиндеги болуп **аткаруучу** же **жуктөл ч модуль** деп аталат. Мисалы: Эки сандын суммасын чыгаруу.

```
/* эки сандын суммасы */
#include <iostream>
main ( )
{
    int a, b, sum;
    std::cout<<"Эки бүтүн санды киргиз ";
    std::cin>>a>>b;
    sum=a+b;
    std::cout<< " a+b=", sum;
}
```

Жыйынтык: Эки бүтүн санды киргиз

3,5

a+b=8

### Математикалык стандарттык функциялар

sin(x) - C++-нус (аргументи радиан менен);

cos(x) - косинус (аргументи радиан менен);

tan(x) - тангенс (аргументи радиан менен);

asin(x) - арксинус (жыйынтык радиан менен);

acos(x) - арккосинус (жыйынтык радиан менен);

atan(x) - арктангенс (жыйынтык радиан менен);

$\sinh(x)$  - гиперболикалык синус;  
 $\cosh(x)$  –гиперболикалык косинус;  
 $\tanh(x)$  - гиперболикалык тангенс;  
 $\log_{10}(x)$  - ондук логарифм;  
 $\text{pow}_{10}(x)$  -10 санын  $x$  даражасына көтөрүү;  
 $\log(x)$  - натуралдык логарифм;  
 $\exp(x)$  –  $e$  санын  $x$  даражасына көтөрүү;  
 $\text{sqrt}(x)$  –  $x$  санынан квадраттык тамыр чыгаруу;  
 $\text{pow}(x,y)$  -  $x$  санын  $y$  даражасына көтөрүү;  
 $\text{fabs}(x)$  – кош тактыктагы  $x$  санынын абсолюттук чоңдугу;  
 $\text{abs}(x)$  -  $x$  бүтүн санынын абсолюттук чоңдугу.

Математикалык стандарттык функцияларды колдонуш үчүн программага `#include <math.h>` сабын жазуу керек

Мисалы:

```

#include <iostream>
#include <math.h>
using
main()
{
  int x , y;
  x = 2;
  y = 3;
  return pow(x,y);
}

```

Жыйынтык: 2 нин 3-даражасы барабар 8

### Өзгөрмөлөр жана турактуулар.

Өзгөрмө жана турактуулардын негизин чоңдук түшүнүгү түзөт. Чоңдуктар программалоо тилинде абдан көп кезиге турган термин.

**Чоңдуктар** – бул кандайдыр бир же сандык, же символдук, же логикалык маанилерди кармаган түшүнүк. Чоңдуктар эки түргө бөлүнөт: өзгөрмө жана турактуу.

**Турактуулар** - Си тилинде турактуулар типтерге бөлүнөт: бүтүн типтүү турактуу(integer), символдук типтеги(char-character) турактуу, калкыма же өзгөрүлмө чекитүү(floating point) турактуу, саналуучу(enumeration) турактуу жана сап(string) турактуусу.

Бүтүн типтүү турактуулар төмөндөгү таблица менен берилди.

Тиби	Өлчөмү	Маанилер аралыгы
integer	2 байт	-32768ден 32767ге чейин
unsigned integer	2 байт	0дөн 65535ке чейин
long integer	4 байт	-2147483648ден 2147483647ге чейин

unsigned long integer	4 байт	0дөн 4294967295ке чейин
-----------------------	--------	-------------------------

Бүтүн типтүү турактуулар кыскача (int) сөзү менен программанын текстинде ондук, сегиздик жана он алтылык эсептөө системасында жазылышы мүмкүн. Турактуулар узун бүтүн турактуулар да боло алышат. Узун бүтүн турактуулар татаал эсептөөлөрдү жүргүзүүдө колдонулат.

Калкыма же өзгөрүлмө чекитүү(floating point) турактуулар төмөндөгү таблица менен берилди.

Тиби	Өлчөмү	Маанилер аралыгы
float	4 байт	3.4E-38ден 3.4E+38ге чейин
double	8 байт	1.7E-308ден 1.7E+308ге чейин
long double	8 байт	1.7E-308ден 1.7E+308ге чейин

*Калкыма чекитт турактуулар:* ондук бөлчөктүн бүтүн бөлүгүнөн анын бөлчөк бөлүгүн ажыратуучу чекиттин абалын өзгөртсөк ал калкыма чекит деп аталат. Мисалы: 24.0 саны берилсе анны калкыма чекиттүү кылып жазсак болот 2.4E+1 же 240.0E-1 ж.у.с.

Символдук типтеги турактуулар.

Бул типтеги турактуулар тырмакча( ' ) белгисине алынып жазылат ('a', '2') жана ASCII коддоо таблицасындагы баардык символдор кирет. Бул символдор шарттуу түрдө көрүнүүчү жана көрүнбөөчү болуп эки топко бөлүнөт.

Көрүнүүчү символдук турактууларды экранга же кагазга чыгарууга болот. Көрүнбөөчү символдук турактуулар графикалык көрүнүшкө ээ болбойт. Мисалы: сапты которуу символу, үн чыгаруу символу ж.у.с.

ASCII коддоо таблицасында 0 дон 31 ге чейинки номердеги символдор башкаруучу символдор болуп эсептелет.

Бул башкаруучу символдор Си программасында көрүнбөс символ катары колдонулат.

Буларды кээде башкаруучу коддор деп да аташат. Бул башкаруучу коддор \ белгисинен кийин жазылат. Мисалы:

\a үн чыгарууну берет

\b бир кадамга артка аракетин аткарат

\f жаңы бетке өтөт

\n сапты которот

\t горизонталдык таблицаны жүргүзөт

\v вертикалдык таблицаны жүргүзөт

Саналуучу(enumeration) турактуулар маңызы боюнча бүтүн типтүү турактуулар болушат. Мисалы:

enum

{kyn,dyishondy,sheishenbi,sharshenbi,beishenbi,juma,ishenbi,jekshen bi};

Бул баяндоодо күн деген турактуу 0 турактуусун билдирет, dyishondy- деген турактуу 1 турактуусун билдирет, sheishenbi- деген турактуу 2 турактуусун билдирет, sharshenbi- деген турактуу 3 турактуусун билдирет, beishenbi- деген турактуу 4 турактуусун билдирет, juma- деген турактуу 5 турактуусун билдирет, ishenbi- деген турактуу 6 турактуусун билдирет, jekshenbi- деген турактуу 7 турактуусун билдирет.

Сап турактуулары экиден ашык символдордун удаалаштыгынан турат жана кош тырмакчага(“) алынып жазылат

Мисалы; “C++ тили”, “ПИН-22 тайпасы”.

**Өзгөрмөлөр** - бул программанын иштөө процесинде өзүнүн маанисин өзгөртүүчү чондукту айтабыз. Өзгөрмө ысымга жана мааниге ээ болот. Өзгөрмөлөрдүн ысымдары латын тамгаларынан, цифралардан жана «\_» төмөнкү сызык белгисинен турат. Ысым сөзсүз түрдө латын тамгасынан башталышы керек. Өзгөрмөнүн ысымы анын мааниси жайгашкан эстеги орунду аныктайт. Өзгөрмө статикалык жана динамикалык болуп эки түргө бөлүнөт.

Статикалык өзгөрмөлөр программанын аткарылыш процессинде учурдагы эстен алган ордун турактуу кармап турат жана программа аткарылып бүткөнчө өзүн жоготпойт. Динамикалык өзгөрмөлөр-деп программанын аткарылыш процессинде бирде пайда болуп, кайра жоюлуп туруучу өзгөрмөнү түшүнөбүз.