

**СУЛЕЙМАН ДЕМИРЕЛ УНИВЕРСИТЕТИ**

**Жапаров М. К.**

**C++ БАҒДАРЛАМАЛАУ ТІЛІ**

**Оқу-әдістемелік құралы**

**Қаскелен, 2015**

**ӘОЖ 004004. 4(075.8)**  
**ҚБЖ 32.973.26-018 я73**  
**Ж 30**

Сулейман Демирель атындағы университеттің Оқу-әдістемелік кеңесімен ұсынылған №1 хаттама 25.08.2015 ж.

Пікір жазғандар:

Кожухметов К., PhD, ассистент профессор, [Қазақ-Британ техникалық университет](#)  
*Амиргалиев Е. Н. д.т.н., профессор, Сулейман Демирель атындағы университет*

Жапаров М. К.

Ж30 С++ Бағдарламалау Тілі: Оқу-әдістемелік құралы. Жапаров М. К. —  
Каскелен, Сулейман Демирель атындағы университет 2015. — 87 б. — қазақ тілінде.

ISBN 978-601-7537-21-0

«С++ Бағдарламалау Тілі:» оқу-әдістемелік құралы 5В03000 — «Ақпараттық жүйелер» және 5В040000 — «Есептеу техникасы және бағдарламалық қамтамасыз ету» мамандықтарының бакалавриат студенттеріне арналған.

**ӘОЖ 004004. 4(075.8)**  
**ҚБЖ 32.973.26-018 я73**

© Сулейман Демирель атындағы, 2015  
© Жапаров М. К.

ISBN 978-601-7537-21-0

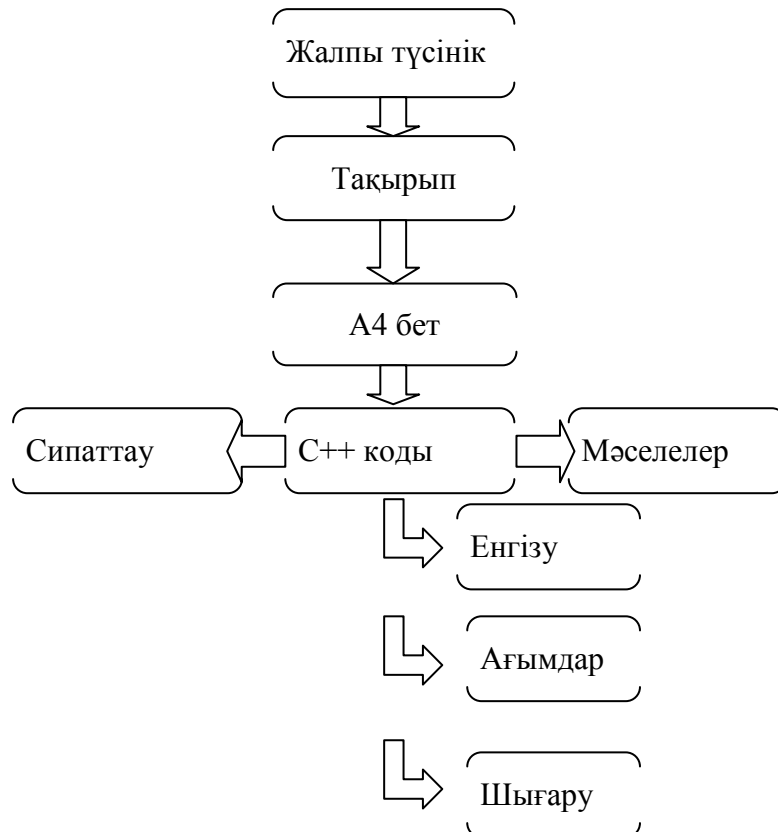
## МАЗМҰНДАМА

Кіріспе	5
1. Қарапайым бағдарламалар ++	7
2. Компиляция және орындау	8
3. Комментарий	9
4. Енгізу/шығару (input/output)	10
5. Мәліметтер түрі (Data types)	11
6. Integer"-дің (бүтін санның) мәліметтер түрі	12
7. Нақты сандардың (real numbers) мәліметтер түрі	13
8. Логикалық (bool) мәліметтер түрі	14
9. Литер (character)	15
10. Түрлендіру (type casting)	16
11. Айнымалылар шегі	17
12. Тұрақты айнымалы (constants)	18
13. Анықтама (define)	19
14. Тізім (enum)	20
15. Арифметикалық операторлар (arithmetic operators),	21
16. Қысқа жазу операторлары (short writing operators)	22
17. Теңдік және теңсіздік операторлары	23
18. Басымдылық операторы (operator precedence)	24
19. кілтсөздер (keywords)	25
20. Ауысу коды (Escape code)	26
21. егер (if) кілтсөзі	27
22. if...else құрылымы	28
23. "=" және "==" айырмашылығы	29
24. "?" кілтсөзді шартты операторы	30
25. бағдарламаларда файлды қолдану	31
26. "for" циклі	32
27. "while" циклі	33
28. "do...while" циклы	34
29. break (тоқтату)	35
30. continue (жалғастыру)	36
31. switch (қосқыш)	37
32. exit (шығу)	38
33. pointer (сілтеуші)	39
34. кері қайтару операторы "*" (deference operator "*")	40
35. pointer (сілтеуіш) көлемі (size of pointer)	41
36. pointer-ге pointer (pointer to pointer)	42
37. массив (array)	43
38. массивтегі циклдар (loops in array)	44
39. көпөлшемді массив (multidimensional array)	45
40. сілтеуіштер мен массивтер арасындағы байланыс	46
41. динамикалық жады (dynamic memory)	47
42. вектор (vector)	48

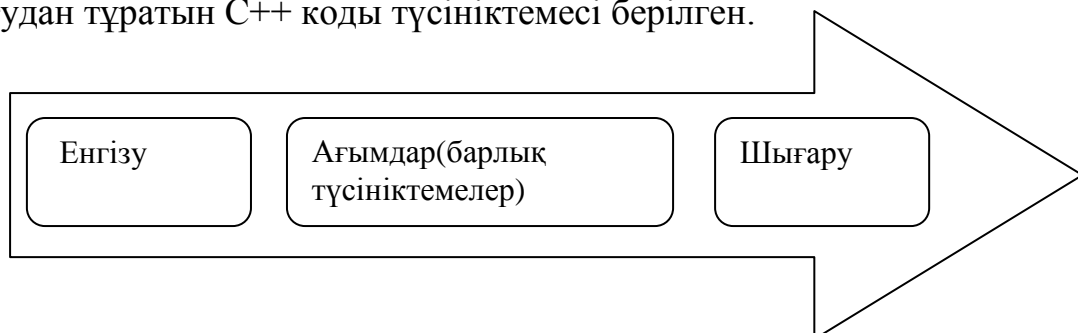
43. вектордағы итераторлар	49
44. векторға арналған кейбір алгоритмдер	50
45. әріптер тізбегі	51
46. Әріптер тізбегінің көлемі	52
47. Буфер	53
48. жалпы функциялар	54
49. : String классы	55
50. Жолдарды оқу	56
51. Өзгерту	57
52. Кейбір string функциялары	58
53. Функция	59
54. “void” типі жоқ функциялар	60
55. Функциядағы шектемелер	61
56. Алдын-ала хабарландыру	62
57. Сыртқа функция	63
58. Есім кеңістігі (namespace)	64
59. Аргументтер тасымалдау	65
60. Класс	66
61. Рұқсат спецификаторы	67
62. Қолжетімділік функциясы	68
63. Конструктор	69
64. Конструктордың артуы	70
65. Стандартты конструктор	71
66. Деструктор	72
67. Сыртқы класс	73
68. Класқа нұсқау	74
69. “this” нұсқаулығы	75
70. Қайта іске қосу операторлары	76
71. Унарлы операторды арттыру	77
72. Бинарлы операторды арттыру	78
73. Енгізу/шығару операциясын арттыру	79
74. Тату функция	80
75. Тату класс	81
76. Мұрагерлік	82
77. Қорғалған мүше	83
78. Мұрагерлік байланыс	84
79. Алынған кластың инициализациясы	85
80. Көптік мұрагерлік	86
Сілтемелер	87

## Кіріспе

Кітап туралы жалпы түсінік 1-суретте бейнеленген. Мұнда ауқым, сілтеме тағы басқалары сияқты ортақ тақырыпты көрсететін негізінен C++ бағдарламалау тілі болып табылады. Әрбір ортақ тақырып қосалқы тақырыптардан тұрады, және олардың барлығы тақырыптар тізімінде тұрады. Бұл тақырыптың әрбір элементінің өз веб-беттері бар, олар 3 бөлімнен құралады: сипаттау, C++ коды және тапсырма.



Сипаттау бөлімі сол тақырыпты қысқа әрі бірегей етіп анықтап түсіндіреді. Бұдан кейін 2-суретте көрсетілгендей бөлім бұтақтары кіру, процесс және шығудан тұратын C++ коды түсініктемесі берілген.



3-суретте C++-тегі Microlearning кейбір әдеттегі мысалдарының коды және нәтижесімен бейнеленген.

**Topic : continue**

The continue causes to pass rest of loop and go to next iteration.

	C++ CODE	RESULT
Сипаттау	<pre> 1 #include &lt;iostream&gt; 2 using namespace std; 3 4 int main() 5 { 6 int n1; 7 cin &gt;&gt; n1; 8 for (int i=n1; i&gt;0; i--) 9 { 10 if (i==4) 11 { 12 cout &lt;&lt; "Continue "; 13 continue; 14 cout &lt;&lt; "STOP "; 15 } 16 cout &lt;&lt; i &lt;&lt; " "; 17 } 18 return 0; 19 } </pre>	<pre> Try 1: 9 9 8 7 6 5 Continue 3 2 1 Try 2: 11 11 10 9 8 7 6 5 Continue 3 2 1 </pre>
C++ кодының нәтижесі		
Мәселелер		

**Problems**

- Write a C++ program that prints all odd numbers till given number

## 1. Тақырып: C++тегі қарапайым бағдарламалар

Төменде келтірілген C++-тегі қарапайым бағдарлама. Әрбір C++ бағдарламасында `int main() {}` функциясы болады. Бұл пайдаланушының кітапхананысы `"# include"` кілт сөзі одан соң кітапхана есімін енгізу арқылы іске қосылады. Сонымен қатар C++-те стандартты кітапхана бар. Бұл функцияларды қолдану үшін әрбір функцияға арнап `"std::"` кілт сөзін енгізуді талап етеді. Алдағы уақытта оңай жолын қолдану көрсетіледі.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	First program
2	<code>int main()</code>	
3	<code>{</code>	
4	<code>std::cout&lt;&lt;"First program";</code>	
5	<code>return 0;</code>	
6	<code>}</code>	

### Тапсырма

1.Экранға "Бұл екінші бағдарлама" шығатындай етіп C++ бағдарламасын жазыңыз.

2. Экранға "Қазақстанға қош келдіңіз" шығатындай етіп C++ бағдарламасын жазыңыз.

## 2. Тақырып: Компиляция және орындау

C++-те бағдарлама жазудың екі жолы бар: бірі қолданушының графикалық интерфейсі (graphical user interface, GUI) арқылы (көп жағдайда Integrated Drive Electronics, IDE деп аталады), екіншісі қара экран (black screen) арқылы (басқаша command, prompt, console, terminal).

Екінші жолының алгоритмі:

1. Папка құру, C++ кеңейткіші бар файл құрып, мәтіндік редактор арқылы ашу (Windows операциялық жүйесінде notepad);
2. C++ кодын жазу (көшірме бола алады);
3. Қара экранды ашу (Windows операциялық жүйесінде Windows белгісі + R пернелерін басып, cmd сөзін жазу арқылы іске асырылады);
4. Cd қолданушы командасы арқылы керекті папканы табу;
5. Қара экранда "g++name\_of\_file.cpp" жазып, Enter пернесін басу;
6. Қате болмаған жағдайда name\_of\_file.exe немесе a.exe шығуы тиіс;
7. Мына файлды жазып, Enter пернесін басыңыз.

	C++ коды	Нәтиже
1	#include<iostream>	Second Program
2	Int main()	
3	{	
4	std::cout<<"Second Program";	
5	return 0;	
6	}	

### Тапсырма

- 1.Экранға "Hello"сөзі шығатындай етіп C++ бағдарламасын жазыңыз.



### 3. Тақырып: Комментарий

C++ бағдарламасында комментарий қоюдың екі жолы бар:

1. Жалғыз жолды комментарий - //

2. Көп жолды комментарий - / \* ...

.....

..... \* /

Көбінесе жаңадан бастап жүрген бағдарламашылар өздерінің бағдарламаларында комментарий жазбайды. Уақыт өте келе тәжірибе барысында комментарийдің керек екенін ұғынады. Себебі, кез келген адамның ұмытарлық қасиеті бар.

	C++ коды	Нәтиже
1	//Comments in C++	Try 1:
2	#include<iostream>	Enter an integer: 5
3		5 is odd
4	int main()	
5	{	Try 2:
6	int i;//single-line comment	Enter an integer: 6
7	/*Read the number	6 is even
8	to be tested whether odd or oven	
9	*/	
10	std::cout<<"Enter an integer";	
11	std::cin>>i;	
12	//check if number i is even or odd	
13	if ((i%2)==0)	
14	std::cout<<i<<"is even";	
15	else	
16	std::cout<<i<<"is odd";	
17	return 0;	
18	}	

#### Тапсырма

1. Берілген санның 3-ке бөлінетінін тексеретін комментарийлары бар C++ бағдарламасын жазыңыз.

2. Комментариялары бар екі "float" санының қосындысы болатын C++ бағдарламасын жазыңыз.

#### 4. Тақырып: Енгізу/шығару (input/output)

C++ бағдарламасында ең көп қолданылатын екі енгізу/шығару (input/output) операторлары бар: cin/cout. Енгізу/шығару (input/output) операторларын қолдану үшін iostream кітапханасын қолдану керек. C++ бағдарламасында стандартты кітапхана бар. Аталған кітапханада көптеген дайын функцияларды қолдануға болады. Ол үшін std кілт сөзін (әдетте namespace деп аталады) қолдану керек.

	C++ коды	Нәтиже
1	#include<iostream>//library for input/output	Try 1:
2	int main()	3
3	{	3
4	int i;//new integer is defined	
5	std::cin>>i;//reading integer from keyboard	Try 2:
6	std::cout<<i;showing integer to display	25
7	return 0;	25
8	}	

#### Тапсырма

1. Екі "integer" санын және олардың көбейтіндісін алатын C++ бағдарламасын жазыңыз.

2. Екі "double" санын және олардың бөліндісін беретін C++ бағдарламасын жазыңыз.

3. Бір "integer" бір "double" сандарын және олардың қосындысын көрсететін C++ бағдарламасын жазыңыз.

## 5. Тақырып: Мәліметтер түрі (Data types)

C++ бағдарламасында компьютер жадында айнымалыларды сақтауға арналған негізгі мәліметтер түрі бар. Компьютер жадында мәліметтер byte түрінде сақталады.

Мәліметтер түрінің мысалдары: Int, long, char.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2		Enter number: 5
3	<code>int main()</code>	Enter a character: d
4	<code>{</code>	5 d
5	<code>int a;</code>	Try 2:
6	<code>char c;</code>	Enter number:12
7	<code>std::cout&lt;&lt;"Enter number:";</code>	Enter a character: Z
8	<code>std::cin&gt;&gt;a;</code>	12 Z
9	<code>std::cout&lt;&lt;"Enter a character:";</code>	
10	<code>std::cin&gt;&gt;c;</code>	
11	<code>std::cout&lt;&lt;a&lt;&lt;" "&lt;&lt;c;</code>	
12	<code>return 0;</code>	
13	<code>}</code>	

### Тапсырма

1. Екі "integer" санын және олардың қосындысын алатын C++ бағдарламасын жазыңыз.

2. Екі "double" санын және олардың қосындысын беретін C++ бағдарламасын жазыңыз.

3. Үш "character" беретін және бір жолға шығаратын C++ бағдарламасын жазыңыз.

## 6. Тақырып: "Integer"-дің (бүтін санның) мәліметтер түрі

Integer – айнымалы, бүтін санды көрсету үшін қолданылады. C++ бағдарламасында int, short, long түрлері бар. Сонымен қатар таңбалы (signed), таңбасыз (unsigned) бола алады.

Таңбасыз деген оң айнымалылар. Өздігінен Integer таңбалы болып келеді. Сонымен қатар C++ бағдарламасында алдын ала бір жолда белгіленген int a,b,c түрлері болады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter numbers:20 40
3	int main()	Enter long numbers:
4	{	1000 1
5	int a1, a2;	60 1001
6	long l1,l2;	Try 2:
7	unsigned short s1;	Enter numbers:1 2
8	std::cout<<"Enter number:";	Enter long numbers:
9	std::cin>>a1>>a2;	100000
10	std::cout<<"Enter long number";	1
11	std::cin>>l1>l2;	3 100001
12	std::cout<<a1+a2<<" " <<l1+l2;	
13	return 0;	
14	}	

### Тапсырма

1. Екі "short integer" сандарының бөліндісін табатын C++ бағдарламасын жазыңыз.

2. 3 "signed" 3 "long integer" сандарының көбейтіндісін табатын C++ бағдарламасын жазыңыз.

3. "long" және "integer" сандарының көбейтіндісін табатын C++ бағдарламасын жазыңыз.

## 7. Тақырып: Нақты сандардың (real numbers) мәліметтер түрі

"Float", "double" түріндегі айнмалылар нақты сандарды сақтау үшін қолданылады. Мысалға, "+3.5", "3.5" және "3.500". Барлығы да сол мәнде ұсынылады. C++ бағдарламасында нақты сандарды белгілеу үшін "," қолданылмайды, орнына "." белгісі қолданылады.

Нақты сандар, сонымен қатар, ғылыми белгілеу жүйесімен беріледі. Мәселен, 0.35 саны "+3.5e+01" жазылуы мүмкін (яғни  $0.35 \times 10$ ).

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter numbers: 3.5 6.7
3	int main()	Enter numbers:4.777-
4	{	9.76
5	double a1,a2;	10.2-4.983
6	float 11,12;	Try 2:
7	std::cout<<"Enter numbers:";	Enter numbers:1.1 2.22
8	std::cin>>a1,a2;	Enter numbers:4-9
9	std::cout<<"Enter numbers:";	3.32-5
10	std::cin>>11>>12;	
11	std::cout<<a1+a2<<" "<<11+12;	
12	return 0;	
13	}	

### Тапсырма

1. Екі "double" санының көбейтіндісін табатын C++ бағдарламасын жазыңыз.
2. Екі "float" санының бөліндісін табатын C++ бағдарламасын жазыңыз.
3. Үш "float" санының қосындысын ғылыми белгілеу жүйесімен көрсететіндей C++ бағдарламасын жазыңыз.

## 8. Тақырып: Логикалық (bool) мәліметтер түрі

Логикалық (bool) мәліметтер түрі тек екі мәнге ие: шын (true) және өтірік (false). Яғни C++ бағдарламасында шын (true) 1-ге, өтірік (false) 0-ге тең.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2		Enter number: 4
3	<code>int main()</code>	More than 0
4	<code>{</code>	1 0
5	<code>bool b1=false,b2;</code>	Try 2:
6	<code>int a1;</code>	Enter number:-4
7	<code>std::cout&lt;&lt;"Enter number:";</code>	0 0
8	<code>std::cin&gt;&gt;a1;</code>	
9	<code>if(a1&gt;0)b1=true;</code>	
10	<code>if(b1)std::cout&lt;&lt;"More than 0\n";</code>	
11	<code>//same as if(b1==true)</code>	
12	<code>b2=0;//same as b2=false</code>	
13	<code>std::cout&lt;&lt;b1&lt;&lt;" " &lt;&lt;b2;</code>	
14	<code>return 0;</code>	
15	<code>}</code>	

### Тапсырма

1. Экранға "101" деп шығаратындай логикалық (bool) мәліметтер түрін пайдаланып, C++ бағдарламасын жазыңыз.

2. Логикалық (bool) мәліметтер түрін қолданып, берілген санның тақ не жұп екенін тексеретін C++ бағдарламасын жазыңыз.

## 9. Тақырып: Литер (character)

C++ бағдарламасында литер мәліметтер жүйесі жеке белгілерді белгілеу үшін қолданылады. Үлкен және кіші әріптер Ағылшын тілі әліппесіндегі белгілерге, сандар "0"... "9", басқа да арнайы \*, @, ! тектес символдарға сай стандартты литерлар бар. Литерлардың жиынтығын ASCII (American Standard Code for Information Interchange) деп атайды.

Символдар бірлік тырнақшалармен белгіленеді ('). Мысалы, C++-те (char c='a'); дұрыс жұмыс істейді, ал (char c="a") компиляциялық қате болып табылады. Символдар (char) сандармен (int) көрсетіледі. (char ол int-тің ішкі жиыны). Мысалға, ASCII-да '9' саны 57 болып белгіленеді, ал 'A' символы 65.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter characters: 5 7
3	int main()	53 55
4	{	5-7=-2
5	char b1, b2;	Try 2:
6	std::cout<<"Enter characters:";	Enter characters: Z A
7	std::cin>>b1>>b2;	90 65
8	std::cout<<int(b1)<<" " <<int(b2)<<"\n";	Z-A=25
9	std::cout<<b1<<"-"<<b2<<"="<<b1-b2;	
10	return 0;	
11	}	

### Тапсырма

1. Берілген integer арқылы литер (character) көрсететіндей C++ бағдарламасын жазыңыз.

2. Литер (character) таңбасын оқып, сан, әріп не арнайы символдар екенін анықтайтын C++ бағдарламасын жазыңыз.

## 10. Тақырып: Түрлендіру (type casting)

Бір int санын екінші бір int санына бөлген кезде жауабы да int саны болады. Жауабы double типінде болу үшін түрлендіруді (type casting) қолдану керек.

Екі жолы бар:

жауабы = (double) (бөлінгіш) / бөлгіш

жауабы = static\_cast <double> (бөлінгіш) / бөлгіш

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter numbers: 5 2
3	int main()	5\2=2
4	{	5\2=2.5
5	Int b1, b2;	5\2=2.5
6	Double d1, d2, d3;	Try 2:
7	std::cout<<"Enter numbers:";	Enter numbers:12 4
8	std::cin>>b1>>b2;	12/4=3
9	d1=b1/b2;	12/4=3
10	d2=(double)(b1)/b2;	12/4=3
11	d3=static_cast<double>(b1)/b2;	
12	std::cout<<b1<<"/"<<b2<<"="<<d1<<"\n";	
13	std::cout<<b1<<"/"<<b2<<"="<<d2<<"\n";	
14	std::cout<<b1<<"/"<<b2<<"="<<d3<<"\n";	
15	return 0;	
16	}	

### Тапсырма

1. Берілген 2 double санын 3 бөліп, мәнді int типінде анықтайтын C++ бағдарламасын жазыңыз.



## 11. Тақырып: Айнымалылар шегі

Айнымалы шегі ауқымды (global) немесе жергілікті (local) бола алады.

Ауқымды айнымалы код басында негізгі денеде (main body) барлық функциялардан тысқары жергілікті айнымалы белгіленгенге дейін не тосқауыл қойылғанға дейін белгіленеді. Егер де жергілікті мен ауқымды айнымалылардың аттары бірдей болған жағдайда "::" кілт сөзі арқылы ауқымдысы қолданылады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2		Enter number: 8
3	<code>int x1=5, x2=10;//global variables</code>	Local x1:8
4		Global x1:5
5	<code>int main()</code>	Global x2:10
6	<code>{</code>	Try 2:
7	<code>Int x1; //local variables</code>	Enter number:14
8	<code>std::cout&lt;&lt;"Enter number:";</code>	Local x1:14
9	<code>std::cin&gt;&gt;x1;</code>	Global x1:5
10	<code>std::cout&lt;&lt;"local x1:"&lt;&lt;x1&lt;&lt;"\n";</code>	Global x2:10
11	<code>std::cout&lt;&lt;"global x1:"&lt;&lt;::x1&lt;&lt;"\n";</code>	
12	<code>std::cout&lt;&lt;"global x2:"&lt;&lt;x2&lt;&lt;"\n";</code>	
13	<code>return 0;</code>	
14	<code>}</code>	

### Тапсырма

1. 3 integer оқытын C++ бағдарламасын жазыңыз. Әрқайсысына мәні 5-ке тең global int x қосыңыз.

## 12. Тақырып: Тұрақты айнымалы (constants)

Тұрақты айнымалылар алдын ала белгілі бір мәнге ие болады, әрі өзгеріске ұшырамайды. Құрылымы:

`const мәлімет_типi айнымалы_аты = мәні;`

Ескерту: тұрақты айнымалы алдын ала белгілену керек.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	compilation error
2		Try 1:
3	<code>int main()</code>	line 5:error:unitialized
4	<code>{</code>	const pi
5	<code>const double pi;//error, it must be initialized</code>	
6	<code>const int x2=5;//no error</code>	
7	<code>int x3;</code>	
8	<code>std::cout&lt;&lt;"Enter radius:";</code>	
9	<code>std::cin&gt;&gt;x3;</code>	
10	<code>std::cout&lt;&lt;"Area:"&lt;&lt;pi*x3*x3&lt;&lt;"\n";</code>	
11	<code>return 0;</code>	
12	<code>}</code>	

### Тапсырма

1. 3 integer оқитын C++ бағдарламасын жазыңыз. Әрқайсысына мәні 5-ке тең тең global int x қосыңыз.
2. C++ бағдарламасын дұрыс жұмыс жасайтындай жазыңыз.
3. Тұрақты айнымалылары бар пирамиданың көлемін табатын C++ бағдарламасын жазыңыз.

### 13. Тақырып: Анықтама (define)

C++ бағдарламасында "#define" атты кілт сөзі тұрақты айнымалыларға өзіндік ат анықтау үшін қолданылады. Басқаша макрос (macros) деп аталады.

Құрылымы:

#define identifier мәні.

Ескерту: көбінесе олимпиадаларда қолданылады. Үлкен проектерде қолданбаған жөн. Командалық жұмыста түсініспеушілік болуы мүмкін.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter radius: 2
3	#define pi 3.14	6.18
4		Try 2:
5	int main()	Enter radius:1
6	{	3.14
7	int x3;//radius	
8	std::cout<<"Enter radius:";	
9	std::cin>>x3;	
10	std::cout<<x3*pi<<"\n";	
11	return 0;	
12	}	

#### Тапсырма

1. Тұрақты анықталған айнымалылары бар шеңбердің ауданын табатын C++ бағдарламасын жазыңыз.

#### 14. Тақырып: Тізім (enum)

Берілген аттарды int мәліметтер типі ретінде тізім (enumeration) қолданылады. Мысалға, аптаның 7 күні. Төмендегідей тізім ретінде жазыла алады:

тізім Апта {ДС=1, СС, СР, БС, ЖМ, СБ, ЖС=0} Осы мысал арқылы оңай түсінетін код жазуға болады.

"Enum" мүшелеріне өздігінен 0,1,2, т.б. мән беріледі. Бірақ алғашқы айнымалыға мән береді. Апта=1 деп жазған кезде қате (compilation error) шығуы мүмкін.

	C++ коды	Нәтиже
1	//in shop at Saturday there is a discount 20%	Try 1:
2	//at Sunday discount 40%	TODAY 0
3	//no discount at other days	20
4	#include<iostream>	Pay: 12
5		Try 2:
6	int main()	TODAY 0
7	{	24
8	enum	Pay:14.4
9	Week{MON=1,TUE,WED,THU,FRI,SAT,SUN=0};	
10	Week w;	
11	w=SUN;//lets assume that today is sunday	
12	std::cout<<"TODAY"<<w<<"\n";	
13	int price;	
14	std::cin>>price;	
15	if(w==SAT)	
16	std::cout<<"Pay:"<<price-price*0.2;	
17	else if(w==SUN)	
18	std::cout<<"Pay:"<<price-price*0.4;	
19	else std::cout<<"Pay:"<<price;	
20	return 0;	
	}	

#### Тапсырма

1. Келтірілген C++ бағдарламасын өзге бір күндермен, басқа мысалмен өзгертіп жазыңыз.

2. Үш түрлі түсі бар (қызыл, сары, жасыл) тізім арқылы бағдарламашамға арналған C++ бағдарламасын жазыңыз.

## 15. Тақырып: арифметикалық операторлар (arithmetic operators)

C++ бағдарламасында бес стандартты арифметикалық операторлар бар:

Операция	Арифметикалық оператор
Қосу	+
Азайту	-
Көбейту	*
Бөлу	/
Қалдық	%

	C++ коды	Нәтиже
1	//Calculator	Try 1:
2		Enter two numbers: 1 5
3	#include<iostream>	Sun:6
4		Try 2:
5	int main()	Enter two numbers:2 3
6	{	Sun: 5
7	int x,y;	
8	std::cout<<"Enter two numbers:";	
9	std::cin>>x>>y;	
10	std::cout<<"Sun:"<<x+y;	
11	return 0;	
12	}	

### Тапсырма

1. Бес операторға арналған калькулятор жазыңыз.

## 16. Тақырып: қысқа жазу операторлары (short writing operators)

Қысқа жазу кодында белгілеу структура операторлары (compound assignment operators) қолданылады. Белгілеу структура операторлары деген құрамында арифметикалық және теңдік (=) операторлары бар операторларды атайды. Мысалға "x=x+5" келесідегідей "x+=5" жазыла алады.

Сонымен қатар өспелі (increase,++) немесе кемімелі (decrease,--) операторларды қолдануға болады. Мысалға, "x=x+1" өрнегін "x++" деп жазуға болады.

	C++ коды	Нәтиже
1	//Calculator	Try 1:
2	#include<iostream>	Enter two numbers: 4 5
3		Sun:9
4	int main()	Try 2:
5	{	Enter two numbers:3 6
6	int x,y;	Sun:9
7	std::cout<<"Enter two numbers:";	
8	std::cin>>x>>y;	
9	x+=y;//same as x=x+y	
10	std::cout<<"Sun:"<<x;	
11	return 0;	
12	}	

### Тапсырма

1. C++ бағдарламасында калькулятор жазыңыз.

## 17. Тақырып: теңдік және теңсіздік операторлары (equality and relation operators)

C++ бағдарламасындағы шешімдерге арналған теңдік және теңсіздік операторлары:

Теңсіздік операторлары	Мағынасы
>	x у-тен үлкен
<	x у-тен кіші
>=	x у-тен үлкен немесе тең
<=	x у-тен кіші немесе тең

Теңдік операторы

==	x у-ке тең
!=	x у-ке тең емес

Ескерту: көбі "if" кілт сөзі арқылы қолданылады.

	C++ коды	Нәтиже
1	//Comparator	Try 1:
2	#include<iostream>	Enter two numbers: 1 2
3		1 is less than 2
4	int main()	Try 2:
5	{	Enter two numbers:2 1
6	int x,y;	2 is less than 1
7	std::cout<<"Enter two numbers:";	
8	std::cin>>x>>y;	
9	if(x>y);	
10	std::cout<<x<<"is more than"<<y;	
11	if(x<y);	
12	std::cout<<x<<"is less than"<<y;	
13	return 0;	
14	}	

### Тапсырма

1. Жоғарыдағы кодты барлық теңсіздік операторларына қолданылатындай өзгертіп жазыңыз.

## 18. Тақырып: басымдылық операторы (operator precedence)

C++ бағдарламасында алгебрадағыдай басымдылық операторы (қандай операторлардың бірінші орындалатыны) бар. Қарапайым алгоритмі:

Алдымен жақша ішіндегісі есептеледі. Егер де бірнеше жақша болса, ішіндегісі бірінші орындалады.

Екінші кезекте көбейту, бөлу, модуль амалдары орындалады. Егер де бірден көп болса, солдан оңға қарай рет-ретімен орындалады.

Ең соңында қосу және алу амалдары орындалады. Сәйкесінше солдан оңға қарай рет-ретімен орындалады.

C++ бағдарламасында басқа типтегі өрнектер берілуі мүмкін. Мәселен, "a>>b+a=d%3+(5==3)+c". Бұл жағдайда a,b,c,d айнымалы болса ғана іске асады.

	C++ коды	Нәтиже
1	//Operator precedence	Try 1:
2	//in comments order numbers of operations	Enter two numbers: 5 7
3	#include<iostream>	192
4		8
5	int main()	Try 2:
6	{	Enter two numbers:6 6
7	int x,y;	360
8	std::cout<<"Enter two numbers:";	9
9	std::cin>>x>>y;	
10	int z;	
11	z=((x-y)+(x*x-y))*(x+y);	
12	//7 1 4 2 3 6 5	
13	std::cout<<z<<"\n";	
14		
15	z=((x==y)+"(x%4))+y;	
16	//5 1 3 2 4	
17	std::cout<<z;	
18	return 0;	
19	}	

### Тапсырма

1. Екі нүкте арасындағы қашықтықты табатын C++ бағдарламасын жазыңыз, <cmath> кітапханасындағы sqrt функциясын қолданыңыз.

2. "z=((x=y)+(x++)) + (x!=y);" өрнегінен қандай жауап шығады? Басымдылық операторын жазыңыз.



## 19. Тақырып: кілтсөздер (keywords)

C++ бағдарламасында анық бірнеше кілтсөздер бар. Айнымалылар аттары кілтсөз бола алмайды. Сонымен қатар C++ бағдарламасында return мен RETURN айырмашылығы бар.

Кілтсөздер тізімі:

asm, auto, bool, case, catch, char, class, const, const\_cast, continue, default, delete, do, double, dynamic\_cast, else, enum, explicit, export, extern, false, float, for, friend, goto, if inline, int, long, mutable, namespace, new, operator, private, protected, public, register, reinterpret\_cast, return, short, signed, sizeof, static, static\_cast, struct, switch, template, this, throw, true, try, typedef, typeid, typename, union, unsigned, using, virtual, void, volatile, wchar\_t, while.

Қосымша:

and, and\_eq, bitand, bitor, compl, not, not\_eq, or, or\_eq, xor, xor\_eq.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1: Compilation error
2		
3	int main()	
4	{	
5	int x,y;	
6	std::cout<<"Enter two numbers:";	
7	std::cin>>x>>y;	
8	int for;	
9	for=x+y;std::cout<<for;	
10	return 0;	
11	}	

### Тапсырма

1. Келтірілген C++ бағдарламасын қатесіз іске қосылатындай етіп жазыңыз.

## 20. Тақырып: Ауысу коды (Escape code)

C++ бағдарламасында ауысу коды (escape code) бар. Ол қосымша символдар мен әрекеттерді көрсету үшін арналған. Құрылымы "/" белгісі мен әріптен тұрады.

\n жаңа сызық  
\r каретканың қайтарымы  
\t tab  
\v вертикал tab  
\b қайтарым бір айқындамаға  
\f шикізаттан(беттің берісі)  
\a дыбыстық сигнал (beep)  
\' бірлік тырнақша (')  
\" екілік тырнақша (")  
\? Сұрақ белгісі (?)  
\\ қарама-қарсы слэш (\)

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter two chars:a W
3	int main()	a?'W Try 2:
4	{	Enter two chars:2 z
5	char ch1,ch2;	2 ? ' z
6	std::cout<<"Enter two chars:";	
7	std::cin>>ch1>>ch2;	
8	std::cout<<"-----\n";	
9	std::cout<<ch1<<"\t \?'\t"<<ch2;	
10	return 0;	
11	}	

### Тапсырма

1. Мәтінді "double quotes"-та шығарып беретін C++ бағдарламасын жазыңыз.

## 21. Тақырып: егер (if) кілтсөзі

Көптеген бағдарламалау тілінде "if" шешімінің кілтсөзі бар. Шартты тексеру үшін қолданылады. Құрылымы:

if (шарт немесе шарттар) C++ коды НЕМЕСЕ (OR)

if (шарт немесе шарттар)

{C++ коды}

Кешенді шарттарға арнап логикалық операторлар қолданылуы да мүмкін.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2		Enter char: a
3	<code>int main()</code>	You entered letter a
4	<code>{</code>	Try 2:
5	<code>char ch1;</code>	Enter char: d
6	<code>std::cout&lt;&lt;"Enter char:";</code>	You do not entered letter
7	<code>std::cin&gt;&gt;ch1;</code>	a
8	<code>if (ch1=='a')</code>	
9	<code>std::cout&lt;&lt;"You entered letter a";</code>	
10	<code>if(ch1!='a')</code>	
11	<code>std::cout&lt;&lt;"You do not entered letter a";</code>	
12	<code>return 0;</code>	
13	<code>}</code>	

### Тапсырма

1. Берілген санның тақ не жұп екенін тексеретін C++ бағдарламасын жазыңыз.

2. Берілген санның 2 немесе 3 саннан (digit) тұратынын тексеретін C++ бағдарламасын жазыңыз. Модуль операторын "%" қолданыңыз.

## 22. Тақырып: if...else құрылымы

C++ бағдарламасында есептерді шығару үшін if...else құрылымы қолданылады. Құрылымы:

if (шарт) код

else if (шарт) код else код

Ескерту: егер де 2 не одан да көп шарттар болса, "{}" жақшалары қолданылады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter char: a
3	int main()	You entered letter
4	{	And it is lower case
5	char ch1;	Try 2:
6	std::cout<<"Enter char:";	Enter char:
7	std::cin>>ch1;	You entered symbol
8	if (ch1=='a' and ch2<='z' )	
9	{	
10	std::cout<<"You entered letter \n";	
11	std::cout<<"And it is lower case";	
12	}	
13	else if (ch1>='A' && ch1<='Z')	
14	std::cout<<"You do not entered capital letter ";	
15	else if (ch1>='0' && ch1<='9')	
16	std::cout<<"You entered digit";	
17	else	
18	{	
19	std::cout<<"You entered symbol";	
20	char ch2=ch1;	
21	}	
22	return 0;	
23	}	

### Тапсырма

1. Берілген нүктенің үшбұрыш ішінде не сыртында жатқанын тексеретін C++ бағдарламасын жазыңыз.

2. Екі тікбұрыштың қайсысы үлкен не тең екенін анықтайтын C++ бағдарламасын жазыңыз.

### 23. Тақырып: "=" және "==" айырмашылығы

Жаңа бағдарламашылар немесе басқа бағдарламалау тілін меңгерген бағдарламашылар оператордағы "=" және "==" айырмашылығын шатастырады.

Көбінесе if құрылымындағы жіберілген қателіктерден болады. Мысалға, C++ бағдарламасында жазылған “if(n=5)” және “if(n==5)” кодтарды салыстырсақ:

Алғашқысында n саны 5-ке теңестіріледі. Соңғысында n саны 5-пен тексеріледі және 0(true) немесе 1(false) кері қайтарады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter two numbers:2 3
3	int main()	2 is not equal to 5
4	{	5 is equal to 5
5	int n1,n2;	
6	std::cout<<"Enter two numbers:";	Try 2:
7	std::cin>>n1>>n2;	Enter two numbers: 9 9
8		9 is not equal to 5
9	if (n1==5)//checks n1 and 5	5 is equal to 5
10	std::cout<<n1<<"is equal to 5\n";	
11	else	
12	std::cout<<n1<<"is not equal to 5\n";	
13	if (n2=5)//assigns n2 to 5	
14	std::cout<<n2<<"is equal to 5\n ";	
15	else	
16	std::cout<<n2<<"is not equal to 5\n";	
17	return 0;	
18	}	

#### Тапсырма

1. Берілген n кібісе (високосный) жыл екенін тексеретін C++ бағдарламасын жазыңыз.

2. Жазылған әріптің үлкен не кіші әріп екенін анықтайтын C++ бағдарламасын жазыңыз.

## 24. Тақырып: "?" кілтсөзді шартты операторы

C++ бағдарламасында if...else құрылымының орнына "?" шартты операторын қолдануға болады. Құрылымы:

шарт ? нәтиже1 : нәтиже2;

Сонымен қатар осы операторды cout ағынында да қолданса болады, ол C++-тің маңызды ерекшелігі болып табылады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2		Enter two numbers: 1 4
3	int main()	4 is maximum
4	{	1 is minimum
5	int n1,n2;	Try 2:
6	std::cout<<"Enter two numbers:";	Enter two numbers: 8 2
7	std::cin>>n1>>n2;	8 is maximum
8	int max;	2 is minimum
9	max = (n1<n2) ? n1:n2;	
10	std::cout<<max<<"is maximum \n";	
11	std::cout<<((n1<n2)? n1:n2)"is minimum ";	
12	return 0;	
13	}	

### Тапсырма

1. Шартты операторлар арқылы берілген үш саннан ең үлкенін табатын C++ бағдарламасын жазыңыз.

## 25. Тақырып: бағдарламаларда файлды қолдану

Көбінесе практикада оқу және жазу үшін файлдарды қолданған жеңіл келеді. Қайта-қайта тексеру үшін айнаымалыларды жазу көп уақытты талап етеді. Сондықтан да бағдарламашылар мәліметтерді файлдан оқып, жауабын экранға шығаруды жөн көреді.

<p>Its structure is:</p> <pre>#include&lt;fstream&gt; using namespace std; int main() { ifstream fin("name_of_file"); ofstream fout("name_of_file"); fin&gt;&gt; some_data;fout&lt;&lt;some_data; fin.close();fout.close();return 0; }</pre>	<p><b>RULES</b></p> <ol style="list-style-type: none"> <li>1.Instead of&lt;iostream&gt;&lt;fstream&gt;is included.</li> <li>2.In folder where program code is input file must be created.</li> <li>3.Instead of cin,cout use fin,fout</li> <li>4.At end close them</li> </ol> <p>Note:do not write in program something like" enter two numbers" because program reads from file.</p>
--	---

Ескерту: std::fin немесе std::cin орнына namespace std жазылуы мүмкін. Осылай тиімдірек.

	C++ коды	Нәтиже
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	<pre>#include&lt;iostream&gt; using namespace std; int main() { int n1,n2; ifstream fin("input.txt"); ofstream fout("output.txt"); fin&gt;&gt;n1&gt;&gt;n2; int max; max = (n1&lt;n2) ? n1:n2; fout&lt;&lt;max&lt;&lt;"is maximum \n"; fout&lt;&lt;((n1&lt;n2)? N1:n2)&lt;&lt;"is minimum"; fin.close(); fout.close(); return 0; }</pre>	<p>Try 1: in input.txt file 3 4 in output.txt file 4 is maximum 3 is minimum</p>

### Тапсырма

1. "digit.txt" файлынан 15 сан оқып, кейін "output.txt" файлына сақтайтын C++ бағдарламасын жазыңыз.

## 26. Тақырып: "for" циклі

"for" циклінің құрылымы:

for (инициализация; шарт; арттыру)

{ C++ коды }

Төрт кезеңнен тұрады. Алғашқы екеуі ";" белгісімен аяқталады. Олар:

1. Алғашқы түрге келтіру. Бір-ақ рет орындалады.
2. Шарт тексеріледі. Дұрыс болған жағдайда цикл әрі қарай жалғасады. Қате болса цикл тоқтайды, код өз ісін доғарады.
3. Код іске асады. Егер де бірнеше шарттан тұрса, "{}" ішіне алынады.
4. Орындалған болса өседі. Цикл екінші кезеңге қайта оралады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	5
3		1 2
4	int main()	2 4
5	{	3 6
6	int n1,n2;	4 8
7	cin>>n1;	5 10
8	for(int i=1; i<=n1; i++)	Try 2:
9	{	3
10	cout<<i<<" " <<i*2;	1 2
11	cout<<end;//newline same as cout<<"\n";	2 4
12	}	3 6
13	return 0;	
14	}	

### Тапсырма

1. Алғашқы n санның жалпы қосындысын табатын C++ бағдарламасын жазыңыз.
2. 0-ден 1-ге дейінгі сандарды көрсететін C++ бағдарламасын жазыңыз.
3. Барлық символдарды басып шығаратын C++ бағдарламасын жазыңыз. Символдар – ASCII кодтағы сандар.



## 27. Тақырып: "while" циклі

"while" циклінің құрылымы:

```
while (шарт(тар) ақиқат) { код }
```

"while" циклында қойылған шарттар өтірікке тексеретін немесе өспелі (кемімелі) оператор болуы керек. Олай болмаған жағдайда цикл шексіз цикл болады.

"for" мен "while" арасындағы айырмашылық: "for" циклінде кезеңдер саны белгілі, ал "while" циклінде кезеңдер саны белгісіз және шартқа байланысты.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	135
3		531 is not palindrome of
4	int main()	135
5	{	Try 2:
6	int n1,n2,n3=0;	135531
7	cin>>n1;	135531 is palindrome of
8	n2=n1;	135531
9	while(n1!=0)	Try 3:
10	{	1234567654321
11	n3=n3*10+(n1%10);	-1126087180 is not
12	n1=n1/10;	palindrome of 2147483647
13	}	(NOTE here error appears
14	if(n3==n2)	because given number is
15	cout<<n3<<"is palindrome of "<<n2;	more than long int range)
16	else	
17	cout<<n3<<"is not palindrome of "<<n2;	
18	return 0;	
19	}	

### Тапсырма

1. Берілген n саны құрамындағы цифрлар санын есептейтін C++ бағдарламасын жазыңыз.

2. Берілген n саны жай сан екенін тексеретін C++ бағдарламасын жазыңыз.

3. Берілген n санының барлық бөлгішін табатын C++ бағдарламасын жазыңыз.

## 28. Тақырып: "do...while" циклы

"do...while" циклының құрылымы:

```
do { код } while (шарт[тар] ақиқат);
```

"while" циклы сияқты жұмыс істейді. Егер де бағдарламашы шартты жалғанға өзгертуді ұмытып кеткен жағдайда, шексіз цикл пайда болуы мүмкін.

"do...while" мен "while" циклдерінің басты айырмашылығы кей кездері "do...while" циклында 1 қадам артық болуы мүмкін. Себебі, бірінші орындауды жүзеге асырады, кейін шартты тексереді.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	15
3		AAAAA
4	<code>int main()</code>	AAAAA
5	<code>{</code>	
6	<code>long int n1,n2,n3=0;</code>	Try 2:
7	<code>cin &gt;n1;n2=10;do</code>	5
8	<code>{</code>	A
9	<code>cout&lt;&lt;"A";</code>	
10	<code>n2++;</code>	
11	<code>}while (n2&lt;n1);</code>	
12		
13	<code>cout&lt;&lt;endl;</code>	
14	<code>n2 = 10;</code>	
15	<code>while(n2&lt;n1)</code>	
16	<code>{</code>	
17	<code>cout&lt;&lt;"A";</code>	
18	<code>n2++;</code>	
19	<code>}</code>	
20	<code>return 0;</code>	
21	<code>}</code>	

### Тапсырма

1. Берілген n санына дейін барлық жай сандарды шығарып беретін C++ бағдарламасын жазыңыз.

2. Берілген n санына дейін төмендегідей шығарып беретін C++ бағдарламасын жазыңыз.

1

22

333

## 29. Тақырып: break (тоқтату)

"Break" командасы циклды аяқтау үшін және қойылған шарт дұрыс бола тұра циклдан шығу үшін қолданылады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	8
3		8 7 6 5 4 Break
4	<code>int main()</code>	Try 2:
5	<code>{</code>	11
6	<code>int n1;</code>	11 10 9 8 7 6 5 4 Break
7	<code>cin &gt;n1;</code>	
8	<code>for( int i=n1; i&gt;0; i--)</code>	
9	<code>{</code>	
10	<code>cout&lt;&lt;i&lt;&lt;" ";</code>	
11	<code>if(i==4);</code>	
12	<code>{</code>	
13	<code>cout&lt;&lt;"Break";</code>	
14	<code>break;</code>	
15	<code>cout&lt;&lt;"STOP";</code>	
16	<code>}</code>	
17	<code>}</code>	
18	<code>return 0;</code>	
19	<code>}</code>	

### Тапсырма:

1. "Break" командасы арқылы берілген сандардың ішінен жай сандарды табатын C++ бағдарламасын жазыңыз.

2. Екі (x, y) сан оқып, арасындағы барлық сандарды көрсететін C++ бағдарламасын жазыңыз. Егер де осы аралықтағы сандардың бірі x және y сандарының қосындысына тең болса, цикл тоқтатылсын (break).

### 30. Тақырып: continue (жалғастыру)

"Continue" циклде кідіріп, келесі (итерацияға) өту үшін қолданылады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	9
3		9 8 7 6 5 Continue 3 2 1
4	<code>int main()</code>	Try 2:
5	<code>{</code>	11
6	<code>int n1;</code>	11 10 9 8 7 6 5 Continue 3
7	<code>cin &gt;n1;</code>	2 1
8	<code>for(int i=n1;i&gt;0;i--)</code>	
9	<code>{</code>	
10	<code>if(i==4)</code>	
11	<code>{</code>	
12	<code>cout&lt;&lt;"Continue";</code>	
13	<code>continue;</code>	
14	<code>cout&lt;&lt;"STOP";</code>	
15	<code>}</code>	
16	<code>cout&lt;&lt;i&lt;&lt;" ";</code>	
17	<code>}</code>	
18	<code>return 0;</code>	
19	<code>}</code>	

#### Тапсырма:

1. Берілген санға дейін барлық тақ сандарды табатын C++ бағдарламасын жазыңыз.

### 31. Тақырып: switch (қосқыш)

"Switch"-ті тиімді қолдану үшін if...else талап етіледі. Құрылымы:

```
switch (variable) {  
  case 1 :  
    // Process for variable = 1  
    break;  
  case 5 :  
    // Process for variable = 5  
    break;  
  default :  
    // Process for all other cases.  
    ....}
```

Ескерту: әрбір кезеңнен кейін break жазылуы тиіс.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	15
3	int main()	AAAAA
4	{	AAAAA
5	int n1;	
6	cout<<"Enter your grade:";	Try 2:
7	cin >n1;	5
8	switch(n1)	A
9	{	
10	case 5:cout<<"Very good";	
11	//break;	
12	case 4:cout<<"Good";break;	
13	case 3:cout<<"Bad";break;	
14	case 2:cout<<"Very bad";break;	
15	default:cout<<"You must enter number between 2 and	
16	5";break;	
17	}	
18	return 0;	
	}	

#### Тапсырма

1. Берілген бүтін санға байланысты ай атын шығаратын C++ бағдарламасын жазыңыз.

### 32. Тақырып: exit (шығу)

"Exit" (шығу) бағдарлама жұмысына шектеу қою үшін қолданылады. Оны кодта қате пайда болған жағдайда қолдануға болады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;cstdlib&gt;</code>	Enter two numbers:
3	<code>using namespace std;</code>	6 3
4		2
5	<code>int main()</code>	Try 2:
6	<code>{</code>	Enter two numbers:
7	<code>int n1,n2;</code>	6 0
8	<code>cout&lt;&lt;"Enter two numbers:";</code>	
9	<code>cin&gt;&gt;n1&gt;&gt;n2;</code>	
10	<code>if(n2==0) exit(1);</code>	
11	<code>int div=n1/n2;</code>	
12	<code>cout&lt;&lt;div;</code>	
13	<code>return 0;</code>	
14	<code>}</code>	

#### Тапсырма:

1. Салыстырмалы қателігі бар квадратты түбірді есептейтін C++ бағдарламасын жазыңыз. Егер салыстырмалы қателік 0-ден кіші болса, бағдарламаға шектеу қою керек.

### 33. Тақырып: pointer (сілтеуші)

Pointer айнымалы, өзінде өзге бір айнымалының мекен-жайын сақтайды. Теледидардың пульті ретінде елестетуге болады. Құрылымы:

мәлімет\_типi \* сілтеуші\_аты;

Айнымалылардың мекен-жайын алу үшін (&) белгісі оператордың мекен-жайы ретінде қолданылады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	12
3	int main()	Address of standart 0xbffc666c
4	{	Value of pointer 0xbffc666c
5	int intStandart;	Address of pointer 0xbffc6668
6	int *intPointer;	
7	cin>>intStandart;	Try 2:
8	intPointer=&intStandart;	22
9	cout<<"Value"<<intStandart<<endl;	Address of standart 0xbffc666c
10	cout<<"Address of standart"	Value of pointer 0xbffc666c
11	<<&intStandart<<endl;	Address of pointer 0xbffc6668
12	cout<<"Value of pointer"<<intPointer<<endl;	
13	cout<<"Address of	Note:address can be changed
14	pointer"<<&intPointer<<endl;	up to system
15	return 0;	
	}	

#### Тапсырма:

1. Double-ды қолдана отырып түзетілген бағдарлама жасайтын C++ бағдарламасын жазыңыз.

2. Integer x мәндерін алатын және ptr1 мен ptr2 жасайтын C++ бағдарламасын жазыңыз. Кейін x, ptr1, ptr2 мәндерін шығарыңыз.

### 34. Тақырып: кері қайтару операторы "\*" (dereference operator "\*")

Pointer деген басқа айнымалының мекен-жайы. Pointer болған айнымалының мәнін білу үшін кері қайтару операторын "\*" (dereference operator "\*") қолданады.

Кері қайтару операторындағы "\*" белгісі мен pointer белгісін шатастырып алмаңыз.

Dereference операторы pointer белгісі болғаннан кейін ғана қолданылады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	24
3	int main()	Value 24
4	{	Value 24
5	int a;	Value after change 48
6	//here * is used for pointer	
7	int * ptr1=&a,*ptr2=&a;	Try 2:
8	cin>>a;	4
9	cout<<"Value"<<a<<endl;	Value 4
10	cout<<"Value"<<*ptr1<<endl;	Value 4
11		Value after change 8
12	//here * is used as dereference	
13	*ptr2=2*a;	
14	cout<<"Value after change"<<*ptr1<<endl;	
15		
16	return 0;	
17	}	

#### Тапсырма:

1. Pointer арқылы екі integer-ді ауыстыратын C++ бағдарламасын жазыңыз.

2. Pointer арқылы екі санның қосындысын көрсететін C++ бағдарламасын жазыңыз.

3. Integer x беретін C++ бағдарламасын жазыңыз. Одан кейін x-ке pointer жасап, оның мәнін өзгертіңіз. Экранға x мәнін, pointer мәнін, dereference қолданылған pointer мәнін көрсетсін.



### 35. Тақырып: pointer (сілтеуіш) көлемі (size of pointer)

Pointer (сілтеуіш) көлемі компьютер құрылымына байланысты 4 байт немесе 8 байт. Pointer-дің көлемі мәліметтер типіне байланысты өзгеріске ұшырамайды. Еске түсірсек, pointer айнымалының тек мекен-жайын сақтайды, мәліметтерін сақтамайды.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	a
3		1.3
4	int main()	1
5	{	4
6	char a;	8
7	drouble b;	4
8	char *ptr1=&a; //pointer for integer double	
9	*ptr2=&b; //pointer for double	Try 2:
10	cin>>a;	Q
11	cin>>b;	2
12	cout<<sizeof(a)<<endl; //size of char	1
13	cout<<sizeof(ptr1)<<endl; //size of pointer	4
14	cout<<sizeof(b)<<endl; //size of double	8
15	cout<<sizeof(ptr2)<<endl; //size of pointer	4
16		
17	return 0;	
18	}	

#### Тапсырма:

1. Int, long, int, float мәліметтер типін және олардың pointer көлемін көрсететін C++ бағдарламасын жазыңыз.

### 36. Тақырып: pointer-ге pointer (pointer to pointer)

C++ бағдарламасында pointer-ге pointer деген түсінік бар. Егер де бір pointer болса, бағдарламашы оған басқа бір pointer арнай алады. Бұл жағдайда сілтемелердің барлық кезеңіне жұлдызша (\*) белгісін қояды.

Құрылымы:

```
мәлімет_типi*сілтеуші1;
```

```
мәлімет_типi **сілтеуші2 = &сілтеуші1; және т.б.
```

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	24
3		Value:24
4	int main()	Address1: 0xbfb4e9ec
5	{	Value2: 0xbfb4e9ec
6	int a; //integer a	Address2:0xbfb4e9e8
7	int *ptr1=&a; //pointer to a	Value3:0xbfb4e9e8
8	int ** ptr2=&ptr1;//pointer to ptr1	Address3:0xbfb4e9e4
9		
10	cin>>a;	Note: the address
11		may
12	cout<<"Value1:"<<a<<"Address1:"<<&a<<endl;	not be same in
13	cout<<"Value2:"<<ptr1<<"Address2:"<<&ptr1<<endl;	testing
14	cout<<"Value3:"<<ptr2<<"Address3:"<<&ptr2<<endl;	
15		
16	return 0;	
17	}	

#### Тапсырма:

1. Char, double, long, int мәліметтер типін және олардың pointer-ге pointer көрсететін C++ бағдарламасын жазыңыз.

### 37. Тақырып: массив (array)

Array (массив) – мәліметтер типі. Құрама айнымалыларды ұқсас типтермен жалғыз атпен индекс қолдану арқылы сақтайды. Индекс деген subscript операторын ([ ]) қолданатын integer параметрі. C++ бағдарламасында индекс 0-ден басталып, массив көлемінен 1-ге кем аяқталады. Индекс массивтің 1-ші i-элементін меншіктеу үшін қолданылады.

Құрылымы:

мәлімет\_типі массив\_аты [массив\_өлшемі];

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	1
3		2
4	int main()	3
5	{	4
6	int a1,a2,a3,a4,a5; //5 integers	5
7	int a[5]; //array for 5 integers	5 4 3 2 1
8		
9	for(int i=0;i<5;i++)	Try 2:
10	cin>>a[i];	5
11		4
12	for(int i=4;i>=0;i--)	3
13	cout<<a[i]<<" ";	2
14		1
15	return 0;	1 2 3 4 5
16	}	

#### Тапсырма:

1. 20 integer-ге арнап массив жасап, 20 integer оқитын, одан кейін тақ, жұп, оң, теріс сандарды және тақ индекстегі санды көрсететін C++ бағдарламасын жазыңыз.

2. 20 double-ға арнап массив жасап, 20 double ішінен максимум, минимумды және солардың индекстерін көрсететін C++ бағдарламасын жазыңыз.

### 38. Тақырып: массивтегі циклдар (loops in array)

Көп жағдайда массив циклдермен (for, while) қолданылады. Олармен операциялар төмендегідей кейіпте болады:

- мәндер ішінен іздеу (x ішінен табу);
- мәндерін есептеу (массивтің орта мәні, қорытындысы);
- массивті өзгерту (кішісінен үлкеніне қарай сұрыптау).

Массивте берілген диапазоннан тыс шығып кету тәрізді қателіктер болуы мүмкін. Ол жоқ индексті шақыру деген сөз. Мәселен, 5 элементтен құралған массив бар делік. Бағдарламашы массив [5] деп шақырады. Бірақ та бесінші элемент массивте [4] болып табылады. Осы сияқты қателіктер болмау үшін көп тәжірибе керек.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	3
3		4
4	int main()	2
5	{	1
6	int size=5;	5
7	int a [size]; //array for 5 integers	1 2 3 4 5
8		
9	for(int i=0; i<size; i++)	Try 2:
10	cin>>a[i];	10
11		9
12	for(int j=i+1;j<size-1;i++)	8
13	{	7
14	for(int j=i+1;j<size;j++)	6
15	if (a[i]>a[j])	6 7 8 9 10
16	swap(a[i],a[j]);	
17	//swap is function for exchange values of two	
18	datas	
19	}	
20		
21	for(int i=0;i<size;i++)	
22	cout<<a[i]<<" ";	
23	return 0;	
	}	

#### Тапсырма:

1. 20 санды үлкенінен кішісіне қарай сұрыптайтын C++ бағдарламасын жазыңыз.

2. Double-дан тұратын үш a,b,c массивтері бар C++ бағдарламасын жазыңыз. a және b массивтері файлдан оқылып, қосындысын c массивінде сақтасын. Шыққан мәнді экранға көрсету керек.

### 39. Тақырып: көпөлшемді массив (multidimensional array)

Көпөлшемді массив деген өзінде массив сақтайтын массив. Басқаша айтқанда, массив ішіндегі массив. Мысалға, матрица тәрізді екі өлшемді массив төменде көрсетілген. Құрылымы:

```
мәлімет_типi массив_аты [жолдар_саны][бағандар_саны] // int a[2][3]
[0] [0] [0][1] [0][2]
[1][0] [1] [1] [1][2]
```

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	2 2
3	<code>#define width 5</code>	1
4	<code>#define height 3</code>	3
5	<code>int main(){</code>	4
6	<code>int a,b;</code>	5
7	<code>cin&gt;&gt;a&gt;&gt;b;</code>	1 3
8	<code>int matrix[a][b]; //multidimensional array</code>	4 5
9		
10	<code>for(int i=0; i&lt;a; i++)</code>	Try 2:
11	<code>for(int j=0;j&lt;b;j++)</code>	3 2
12	<code>cin&gt;&gt;matrix[i][j];</code>	9
13		9
14	<code>for(int i=0;i&lt;a;i++){</code>	9
15	<code>for(int j=0;j&lt;b;j++)</code>	9
16	<code>cout&lt;&lt;matrix[i][j]&lt;&lt;" ";</code>	9
17	<code>cout&lt;&lt;endl; //why we need it here?</code>	9
18	<code>}</code>	9 9
19		9 9
20	<code>return 0;</code>	9 9
21	<code>}</code>	9 1

#### Тапсырма:

1. Екі матрицаны көбейтетін (2x2) C++ бағдарламасын жазыңыз.
2. Көбейту кестесін жазатын C++ бағдарламасын жазыңыз.

#### 40. Тақырып: сілтеуіштер (pointers) мен массивтер (arrays) арасындағы байланыс

C++ бағдарламасында сілтеуіш массивтің бірінші элементінің мекенжайын тең бола алады. Төмендегідей көрсетілуі мүмкін:

```
1 int a [20];
2 int *ptr1 ;
3 ptr1 = a; // ptr1 = &a[0]
```

Бағдарламалаудан кейін келесі элементке ptr1++ арифметикасын қолдану арқылы көшуге болады.

Ескерту: C++ бағдарламасында компилятор автоматты түрде мәліметтер типіне дейін келесі элементке өтеді. Яғни егер массив қосарлы болса, онда сілтеуіш 8 байт болады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	a X!
3	int main()	2
4	{	4
5	char ch[3]={'a','X','!'};	8
6	char*ptr2;	0 2
7	ptr2=ch;	1 4
8	cout<<*ptr2<<" ";	2 8
9	cout<<* (++ptr2)<<" ";	
10	cout<<* (++ptr2)<<" "<<endl;	Try 2:
11	int a[3];	a X!
12	int*b;	4
13	int x;	16
14	b=a; //or b=&a[0];	34
15	cin>>x;	0 4
16	*b=x;	1 16
17	cin>>x;	2 34
18	b++;	
19	*b=x;	
20	cin>>x;	
21	b=a;	
22	*(b+2)=x;	
23	for(int i=0;i<3;i++)	
24	cout<<i<<" "<<a[i]<<endl;	
25	return 0;}	

#### Тапсырма:

1. Сілтеуіш арқылы массивтегі тек жұп сандарды санайтын C++ бағдарламасын жазыңыз. Қолдану керек: for (int \*ptr = a; ptr < a + 3; ptr++), мұндағы a – массив, 3 – массивтің өлшемі.

#### 41. Тақырып: динамикалық жады (dynamic memory)

Көбінесе бағдарламаларда жады көлемі бағдарлама дайын болғанға дейін белгісіз болады. Егер де бағдарламашы жады көлемін орындалып жатқан кезде өзгерткісі келсе, жанасы құрылып, ескісі өшіріледі. Динамикалық жады көлемі қажеттілігіне байланысты өзгереді.

```
int *ptr = new int; // integer-ді динамикалық түрде белгілеп алады
*ptr = 7; // осы integer-ге 7-ні меншіктейді
delete ptr; // ptr мәніне теңестірілген меншікті жадыны жояды
```

Ескерту: операторды жою сілтеуішті жою деген ұғым емес, бірінші жадыны жояды, сілтеуіш тек нұсқайды.

Массивтер үшін new[], delete[] қолданылады.

```
int *ptr = new int[5];
// өлшемі өзгеріп тұратын массивті динамикалық түрде белгілеп алады
*ptr[2] = 7; // массивтің екінші элементін 7-ні меншіктейді
delete[] ptr; // delete-пен шатастырмаңыз.
```

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	4
3	int main()	1
4	{	2
5	int *a=new int;	3
6	cin>>*a;	4
7		1 2 3 4
8	int*b=new int[*a];	
9	for(int i=0;i<*a;i++)	Try 2:
10	cin>>b[i];	3
11	for (int i=0;i<*a;i++)	-1
12	cout<<b[i]<<" ";	-6
13	delete a;	8
14	delete[] b;	-1 -6 8
15	return 0;	
16	}	

#### Тапсырма:

1. Integer n оқып, double-дан көлемі n-нен тұратын динамикалық массив жасап, массивтегі максимал санды табатын, одан соң барлығын өшіріп, қайтадан integer n оқып, long int-тен көлемі n-нен тұратын динамикалық массив жасап, массивтегі минимал санды табатын C++ бағдарламасын жазыңыз.

## 42. Тақырып: вектор (vector)

Вектор динамикалық массив секілді. Вектордың бір-біріне көршілес орналасқан өзіндік элементтер құрылымы бар. Массивтегідей қолжетімді бола алады. Вектордың массивке қарағанда артықшылығы бар, программа орындалып жатқан кезде қажеттілікке байланысты өзгертілуі мүмкін. Құрылымы:

```
vector<мәлімет_типi> вектор_аты; // немесе  
vector< мәлімет_типi > вектор_аты (өлшемі);
```

Ескерту: векторды қолдану үшін кодта `#include <vector>` болу керек.

Вектор көлемін өзгерту үшін бағдарламашы `resize(size)`, `push_bach(жаңа_элемент)` қолдану керек.

Сонымен қатар оңай бағдарламалау үшін стандартты кітапхананы қолдануға болады. Олардың кейбіреулері:

<code>Begin(), end()</code>	Вектордың басы немесе соңы
<code>size()</code>	Вектордағы элементтер санын анықтайды
<code>resize(int)</code>	Вектордың өлшемін арттырады немесе кемітеді
<code>push_back(int), pop_back</code>	Векторға элементті қосады немесе жояды

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;vector&gt;</code>	3
3	<code>using namespace std;</code>	1
4	<code>int main(){</code>	3
5	<code>vector&lt;int&gt;v1;</code>	5
6	<code>int a;</code>	Size 1: 3
7	<code>cin&gt;&gt;a;</code>	Size 2: 5
8	<code>v1.resize(a);</code>	1 3 5 80
9	<code>for(int i=0;i&lt;v1.size();i++)</code>	
10	<code>cin&gt;&gt;v1[i];</code>	Try 2:
11	<code>cout&lt;&lt;"Size 1:"&lt;&lt;v1.size()&lt;&lt;endl;</code>	4
12	<code>v1.push_back(80);</code>	-1
13	<code>v1.push_back(90);</code>	7
14	<code>cout&lt;&lt;"Size 2:"&lt;&lt;v1.size()&lt;&lt;endl;</code>	-4
15	<code>v1.pop_back();</code>	5
16	<code>for(int i=0;i&lt;v1.size();i++)</code>	Size 1: 4
17	<code>cout&lt;&lt;v1[i]&lt;&lt;" ";</code>	Size 2: 6
18	<code>return 0;}</code>	-1 7 -4 5 80

### Тапсырма:

1. `student_GPA` векторын жасап, орташа баллын оқушылар санына байланысты табатын C++ бағдарламасын жазыңыз. Орташа мәнін табыңыз.

2. `n double` санынан тұратын вектор жасап, оларды сұрыптайтын C++ бағдарламасын жазыңыз.



### 43. Тақырып: вектордағы итераторлар

Итератор вектордағы элементтерге оңай шығу жолын көрсетеді. Олар векторлардағы элементтерге және толық векторға да тікелей сілтемелер жасай алады. Құрылымы: `vector<int>::iterator i`; Сілтеуіштегідей (pointer) элементке шығу үшін `*` операторы қолданылады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;vector&gt;</code>	3
3	<code>using namespace std;</code>	1
4		2
5	<code>int main()</code>	3
6	<code>{</code>	1 2 3 80
7	<code>vector&lt;int&gt;v1;</code>	
8	<code>vector&lt;int&gt;::iterator i;</code>	Try 2:
9	<code>int a;</code>	2
10	<code>cin&gt;&gt;a;</code>	-4
11	<code>v1.resize(a);</code>	5
12		-4 5 80
13	<code>for( i=v1.begin();i&lt;v1.end();i++)</code>	
14	<code>cin&gt;&gt;*i;</code>	
15	<code>v1.push_back(80);</code>	
16		
17	<code>for( i=v1begin();i&lt;v1.end();i++)</code>	
18	<code>cout&lt;&lt;*i&lt;&lt;" "</code>	
19	<code>return 0;</code>	
20	<code>}</code>	

#### Тапсырма:

1. Итераторды пайдаланып, вектордағы үлкен элементті табатын C++ бағдарламасын жазыңыз.
2. Вектордағы барлық элементті өшіретін C++ бағдарламасын жазыңыз.

#### 44. Тақырып: векторға арналған кейбір алгоритмдер

Көптеген бағдарламалау тілінде алгоритм арнайы кітапханаларда жазылып қойылған. С++ стандартты кітапханасында векторға арналған алгоритмдер берілген.

Алгоритм	Анықтамасы және мысалы
find – берілген мәнді табады. Егер мән табылған босла, алгоритм итераторды қайтарады, кері жағдайда end-ті қайтарады.	if (find (v1 .begin(), v 1 .end(), 25) == v1 .end())
sort – вектордағы элементтерді сұрыптайды	sort (v1 .begin(), v 1 .end());
reverse – вектордағы элементтердің ретін өзгертеді.	reverse (v1 .begin(), v1.end());
replace – берілген жағдайлардың барлығын басқа жағдайларға өзгертеді	replace(v1.begin(), v1.end(), 'a', 'A');

Ескерту 1: С++-та алгоритмді қолдану үшін кодты #include <algorithm> деп жазу керек.

Ескерту 2: Толығырақ ақпарат үшін гуглдан қараңыз.

	С++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	#include<vector>	3
3	3include<algorithm>	1
4	using namespace std;	2
5	int main()	3
6	{	1 2 3
7	vector<int>v1;	3 2 1
8	int a;	
9	cin>>a;	Try 2:
10	v1.resize(a);	4
11	for(int i=0;i<v1.size();i++)	-4
12	cin>>v1[i];	5
13	sort(v1.begin();v1.end());	-3
14	for(int i=0;i<v1.size();i++)	6
15	cout<<v1[i]<<" ";	-4 -3 5 6
16	cout<<endl;	6 5 -3 -4
17	reverse(v1.begin(),v1.end());	
18	for(int i=0;i<v1.size();i++)	
19	cout<<v1[i]<<" ";	
20	return 0;	
21	}	

#### Тапсырма:

1.Вектордағы белгілі бір санды табатын С++ бағдарламасын жазыңыз.

#### 45. Тақырып: әріптер тізбегі

Әріптер тізбегін (әріптер массивін) string құрады. String мәтін бөлшегі ретінде әріптер ретін құрады. Әріптер ретін аяқталғанын null character белгісі береді ('\0' (кері слэш, нөл)).

```
char str []="SDU";  
char str [] = {'S', 'D', 'U', '\0' };
```

0	1	2	3
S	D	U	/0

Бұл тип Style string деп аталады. C++-те арнайы <string> класы бар.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	Enter name: Askar
3		Bye Askar!
4	int main()	
5	{	Try 2:
6	char init[]="Enter name:";	Enter name: A B
7	char out[]="Bye";	Bye A!
8	char name[25];	
9	cout<<init;	
10	cin>>name;	
11	cout<<out<<name<<"!";	
12	return 0;	
13	}	

#### Тапсырма:

1. Есімін, тегін оқып, бөлек көрсететін C++ бағдарламасын жазыңыз.
2. Студенттің есімін, тегін, жасын, сыныбын оқып, көрсететін C++ бағдарламасын жазыңыз.

#### 46. Тақырып: Әріптер тізбегінің көлемі

Символдардың көлемі: мәтін + 1 (null operator).char str[] "SDU";

0	1	2	3
S	D	U	/0

Бірақ егер біз берілген көлемді [integer] ретінде жарияласақ, онда көлемі толық көрсетіледі.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	Enter name: Askar
3		13
4	int main()	Enter name:
5	{	25
6	char init[]="Enter name:";	Askar\$e^&#@
7	char name[25];	
8	cout<<init;	Try 2:
9	cin>>name;	Enter name: Askar
10		Almaty
11	cout<<sizeof(init)<<endl;	13
12	for(int i=0;i<sizeof(init);i++)	Enter name:
13	cout<<init[i]<<" ";	25
14		Askar\$e^&#@
15	cout<<endl<<sizeof(name)<<endl;	
16	for(int i=0; i<sizeof(name);i++)	Note: symbols can be
17	cout<<name[i];	different
18	return 0;	
19	}	
20		

#### Тапсырма:

1. Есімін оқып, тек дауыстыларды көрсететін C++ бағдарламасын жазыңыз.

#### 47. Тақырып: буфер

Буфер – мәліметтерді уақытша сақтауға арналған жады. C++ бағдарламасында “cin” қолданғанда пайдаланады. Егер пайдаланушы массивке шектен тыс көп мәлімет енгізсе, қателіктер пайда болады. Атауы “a buffer overflow” (сөзбе сөз аударғанда деңгейінен асып төгілу). Ондай жағдайда басқа жадының ішінде қайта жазылу болып жатса жүреді. Бағдарламаның істен шығуына әкеледі. Істен шығу басқа себептерден де болады.

String типін оқу үшін ұсынылатын тәсіл cin-ді пайдалану болып табылады:

```
cin.getline(массив_аты, символдар_саны);
```

Бұл cin.getline () келесі символдар\_саны-1 символ массив\_атына (1 бұл аяқтаушы null!). шақырады. Кез келген артық символдар өшіріледі. Бұл буферді шектен тыс толып кету проблемасынан арылтады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	Enter name: Askar
3		A s k a r \$ e ^ & # @
4	int main()	
5	{	Try 2:
6	char init[]=”Enter name:”;	Enter name: Askar
7	char name[25];	Almaty
8	cout<<init;	A s k a r A l m a t y \$ e ^
9	cin.getline(name,25);	& # @
10		
11	for(int i=0;i<sizeof(name);i++)	Note: symbols can be
12	{	different
13	if((name[i]>=97)&&(name[i]>=122))	
14	cout<<char(name[i]-‘a’+‘A’)<<” ”;	
15	else cout<<name[i]<<” “;	
16	}	
17		
18	return 0;	
19	}	

#### Тапсырма:

1. Есімін, тегін бос орын арқылы бірге оқып, тек есімін көрсететін C++ бағдарламасын жазыңыз.

2. Кез келген бір мәтінді оқып, ондағы үлкен әріптерді кішіге, кіші әріптерді үлкенге ауыстыратын C++ бағдарламасын жазыңыз.

3. Үш қала, олардың халқы санын бір жолда оқып, тек ең үлкенін көрсететін C++ бағдарламасын жазыңыз.

#### 48. Тақырып: жалпы функциялар

C++ бағдарламасы көптеген функцияларды дұрыс қолдану үшін `string`-пен қамтамасыз етеді.

<code>strcat(destination, source)</code>	Бір жолды ( <code>string</code> ) басқасымен қосады;
<code>strncat(destination, source, n)</code>	Бір жолды ұзындығы <code>n</code> -ға тең екіншісімен қосады;
<code>strcmp(destination, source)</code>	Екі жолды салыстырады (тең болса, 0 қайтарады);
<code>strncmp(destination, source, n)</code>	Символдар саны белгілі екі жолды салыстырады (тең болса, 0 қайтарады);
<code>strlen(string)</code>	Жолдың ұзындығын қайтарады (аяқтаушы <code>null</code> -ді есептемегенде);
<code>strcpy(destination, source)</code>	Бір жолды басқасына көшіреді;
<code>strncpy(destination, source, n)</code>	Бір жолды ұзындығы <code>n</code> -ге тең басқа жолға көшіреді;

Ескерту: қолдану үшін кодқа `#include <cstring>` қосу керек.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;cstring&gt;</code>	Enter name: Askar
3	<code>using namespace std;</code>	Askar
4		Ask
5	<code>int main()</code>	A s k a r
6	<code>{</code>	
7	<code>char init[]="Enter name:";</code>	Try 2:
8	<code>char name[25];</code>	Enter name: Askar A Askar
9	<code>cout&lt;&lt;init;</code>	A
10	<code>cin.getline(name,25);</code>	Ask
11		A s k a r A
12	<code>char last[40];</code>	
13	<code>strcpy(last,name);</code>	
14	<code>cout&lt;&lt;last&lt;&lt;endl;</code>	
15		
16	<code>strncpy(last,name,3);</code>	
17	<code>last[3]=0;</code>	
18	<code>cout&lt;&lt;last&lt;&lt;endl;</code>	
19		
20	<code>for(int i=0;i&lt;strlen(name);i++)</code>	
21	<code>cout&lt;&lt;name[i]&lt;&lt;" ";</code>	
22	<code>return 0;}</code>	

#### Тапсырма:

1. Екі мәтін және `n` санын оқып, одан соң осы мәтіндерді `n`-мен және `n`-сіз салыстыратын C++ бағдарламасын жазыңыз.

2. Екі мәтін оқып, жаңадан құрамында оқылған екі мәтін бар C++ бағдарламасын жазыңыз.

#### 49. Тақырып: String классы

Стандартты кітапханада (std) string классы бар. Бұл std::string жолдармен әлдеқайда оңай жұмыс жасайды. Бағдарламашы жолдарды меншіктеу операторлары арқылы өзі меншіктей алады және автоматты түрде керегінше көлемін өзгерте алады. C++ бағдарламасында #include <string> болу керек.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	#include<string>	Enter name: Askar
3	using namespace std;	Hello Askar
4		5
5	int main()	
6	{	Try 2:
7	char init[]="Enter name:";	Enter name: A B Hello
8	string name;	A
9	cout<<init;	1
10	cin>>name;	
11		
12	cout<<"Hello"<<name<<endl;	
13	cout<<name.length();	
14		
15	return 0;	
16	}	

#### Тапсырма:

1. Екі студенттің есімін, GPA көрсеткішін оқитын C++ бағдарламасын жазыңыз. Кейін кімнің GPA көрсеткіші үлкен, соның есімін басып шығаратын болсын.

## 50. Тақырып: Жолдарды оқу

Толық жолды оқу үшін `getline` (`cin`, жолдың\_аты); қолдану керек. Жолдармен жұмыс жасауы массивтерге ұқсас екенін ұмытпаған жөн.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;string&gt;</code>	Enter name: Askar
3	<code>using namespace std;</code>	Hello Askar
4		5
5	<code>int main()</code>	Ask
6	<code>{</code>	
7	<code>char init[]="Enter name:";</code>	Try 2:
8	<code>string name;</code>	Enter name: Askar B
9	<code>cout&lt;&lt;init;</code>	Hello A
10	<code>getline(cin,name);</code>	7
11		Ask
12	<code>cout&lt;&lt;"Hello"&lt;&lt;name&lt;&lt;endl;</code>	
13		
14	<code>cout&lt;&lt;name.length()&lt;&lt;endl;</code>	
15	<code>cout&lt;&lt;name[0]&lt;&lt;name[1]&lt;&lt;name[2]&lt;&lt;endl;</code>	
16		
17	<code>return 0;</code>	
18	<code>}</code>	

### Тапсырма:

1. Есім мен тегін бір жолда оқитын C++ бағдарламасын жазыңыз. Кейін бөлек шығаратындай болсын.

2. "5 + 4 + 2" тәрізді санды және қосу алу операторлары бар арифметикалық жазбаларды оқитын C++ бағдарламасын жазыңыз. Мәнін есептеңіз.



## 51. Тақырып: Өзгерту (Conversion)

Массивтегі әріптер қатарын оңай жолға өзгертуге болады. Меншіктеу операторы арқылы іске асады:

1 string str1;

2 char chr1 []="SDU";

3 std. = chr1 ;

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	#include<string>	Enter name: Askar
3	using namespace std;	Enter name: Askar
4		
5	int main()	
6	{	Try 2:
7	char init[]="Enter name:";	Enter name: A B Enter name:
8	string name,text;	A B
9	text=init;	
10		
11	cout<<text;	
12	getline(cin,name);	
13		
14	cout<<text<<name<<endl;	
15		
16	return 0;	
17	}	

## 52. Тақырып: Кейбір string функциялары

Жолдармен оңай жұмыс істеуге арналған кейбір функциялар:

Name	Аты
Length	Жолдың ұзындығы
clear	Жолды өшіру
append	Жолға қосу
push_back	Жолға символ қосу
insert	Жолға қою
replace	Жолдың бір бөлігінің орнын ауыстыру
find	Жолдың құрамынан табу

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;string&gt;</code>	text1
3		Text2
4	<code>using namespace std;</code>	Text2text1 text1
5		
6	<code>int main()</code>	
7	<code>{</code>	
8	<code>string str1,str2;</code>	
9		
10	<code>cin&gt;&gt;str1&gt;&gt;str2;</code>	
11	<code>swap(str1,str2);</code>	
12	<code>cout&lt;&lt;str1&lt;&lt;str2&lt;&lt;endl;</code>	
13		
14	<code>str1.clear();</code>	
15	<code>cout&lt;&lt;str1&lt;&lt;str2&lt;&lt;endl;</code>	
16	<code>return 0;</code>	
17	<code>}</code>	

### Тапсырма:

1. Екі жолды оқитын C++ бағдарламасын жазыңыз. Кейін ол екеуінен керекті әріпті іздесін.

### 53. Тақырып: Функция

C++ бағдарламасында функция құрылымы:

мәліметтер\_типi функция\_аты (мәліметтер\_типi параметр\_аты) {}

Параметрлері 0-ден керегінше өзгере алады. Параметрлері үтір (,) арқылы ерекшеленеді. Параметрге сай жергілікті айнымалы ретінде функция айнымалыны (аргументті) ала алады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	3 4
3		Area: 12
4	int area(int a,int b)	3.1
5	{	-3.1
6	int ar;	
7	ar=a*b;	Try 2:
8	return ar;	12 -4
9	}	Area: -48
10	double f1(double x)	-3.009
11	{	3.009
12	return -1*x;	
13	}	
14		
15	int main()	
16	{	
17	int a1,a2;double d1,d2;	
18	cin>>a1>>a2;	
19	cout<<"Area:"<<area(a1,a2)<<endl;	
20	cin>>d1;	
21	d2=f1(d1);	
22	cout<<d2;	
23		
24	return 0;	
25	}	

#### Тапсырма:

1. Екі integer саннан ең үлкенін табатын max функциясын жазыңыз.
2. Үш double санның жалпы қосындысын табатын sum функциясын жазыңыз.

#### 54. Тақырып: “void” типі жоқ функциялар

C++ бағдарламашысы void типі белгісін мәліметтер типі ретінде жаза алады. Оның құрылымы:

```
void функция_аты(мәліметтер_типі параметр_аты) {}
```

Массив мәндерін көрсетуге қолданылған бұл функцияларда “void” типі жоқ.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1: Askar
2	#include<string>	Hi Askar
3	using namespace std;	13
4		26
5	void area(string str)	
6	{	
7	cout<<"Hi"<<str<<endl;	Try 2:Arman
8	}	Hi Arman
9	void show()	25
10	{	50
11	int a;	
12	cin>>a;	
13	cout<<a*2<<endl;	
14	}	
15		
16	int main()	
17	{	
18	string str;cin>>str;	
19	area(str);	
20	show();	
21	return 0;	
22	}	

#### Тапсырма:

1. Он студенттің есімін цикл арқылы оқитын, одан соң print функциясы арқылы біріншісін үлкен әріптен жазатын C++ бағдарламасын жазыңыз. Кейін ол екеуінен керекті әріп іздесін.

## 55. Тақырып: Функциядағы шектемелер

Функцияда айнымалылар осы функцияда жергілікті айнымалы бола алады:

```
void function1 (int x, int y) // x және y осында құрылады
{
```

Бұл жердегі x, y және a бүтін сандары жергілікті айнымалы болып табылады.

```
int a=5; // a осында құрылады
} // x, y және a осы жерде жойылады
```

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	2 3
3		function x: 2
4	int area(int x,int y)	function y: 3
5	{	function a: 6
6	int a;	-----
7	a=x*y;	function x: 3
8	cout<<"function x:"<<x<<endl;	function y: 4
9	cout<<"function y:"<<y<<endl;	function a: 12
10	cout<<"function a:"<<a<<endl;	-----
11	cout<<"-----"<<endl;	main x: 6
12	return a;	main y: 12
13	}	main a: 2
14		
15	int main()	
16	{	
17	int a,b;	
18	cin>>a>>b;	
19	int x=area(a,b);	
20	int y=area(3,4);	
21	cout<<"main x:"<<x<<endl;	
22	cout<<"main y:"<<y<<endl;	
23	cout<<"main a:"<<a<<endl;	
24	return 0;	
25	}	

## 56. Тақырып: Алдын-ала хабарландыру

Функция прототипі деген – орындалусыз функцияның хабарлануы. Бағдарламашы функция прототипінің қалай іске асатынын файл басында хабарлай алады. Бұны алдын-ала хабарландыру дейді. Егер алдын-ала хабарландыру кодта қолданылмаса және функция main функциясынан кейін жазылса, бағдарламашы функцияны қолданар кезде қате шығады.

<pre>int main() { int a=sum(3,4); ..... } int sum (int x,int y) { return x+y; } In code in left there is a compilation error</pre>	<pre>int sum(int,int); int main() { int a =sum(3,4); ..... } int sum(int x,int y) { return x+y; } No compilation error</pre>
--	--

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	5 6
3	int area(int,int);//forward declaration,	Area:30
4	//try to write comment//	
5	//to understand and see compilation error	Try 2:
6		12 2
7	int main()	Area: 24
8	{	
9	int a,b;	
10	cin>>a>>b;	
11	int x=area(a,b);	
12	cout<<"Area:"<<x<<endl;	
13	return 0;}	
14		
15	int area(int x,int y)	
16	{return x*y;}	

### Тапсырма:

1. Алдын-ала хабарландыруды пайдаланып, шеңбердің ауданын, периметрін табатын функцияларды жазыңыз.

### 57. Тақырып: Сыртқа функция

C++ бағдарламасында бағдарламашы басқа файлда функция жаза алады және қолдана алады. Сыртқа файлды қолдану үшін ол файлға қосылған болу қажет және .h кеңейтілімі болу керек. Құрылымы:

```
#include "файл_аты.h"
```

Ескерту: C++ бағдарламасында < > және “ ” арасында айырмашылық бар. Стандартты класта < >, өзіндік жергілікті класта “ ” қосылған.

	C++ коды	Нәтиже
1	sum.h	Try 1:
2	int area (int x,int y)	3 4
3	{	Area: 12
4	return x*y;	
5	}	Try:
6	main.cpp	6 9
7	#include<iostream>	Area: 54
8	#include"sum.h"//here external file is included	
9	using namespace std;	
10		
11	int main()	
12	{	
13	int a,b;	
14	cin>>a>>b;	
15	int x=area(a,b);	
16	cout<<"Area:"<<x<<endl;	
17	return 0;	
18	}	

#### Тапсырма:

1. Үш integer ішінен max, min табатын екі функция жазыңыз. Екеуін де test.h файлында сақтап, main.cpp-де қолданыңыз.

## 58. Тақырып: Есім кеңістігі (namespace)

Екі идентификатор немесе функция есімі бірдей болған жағдайда компилятор қате шығарады, себебі мәліметтер жеткіліксіз. Есім кеңістігі барлық идентификаторлар дара болғанда код ауданында анықталады. Өздігінен барлық айнымалылар ауқымды есім кеңістігінде белгілі. Кодтың тәуелсіз екі фрагменті бір-бірімен қатар қолданған кезде ұшырасып қалу мүмкіндігі болған кезде қателіктерге жолықпау үшін C++ бағдарламасы өзіндік есім кеңістігін ұсына алады. Анықталған есім кеңістігінің ішіндегі заттардың барлығы осы кеңістікке жатады. Ауқымды кеңістікке қатысы жоқ.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;namespace n1</code>	5 3
3	<code>{</code>	namespace n1:8 namespace
4	<code>int function(int x,int y)</code>	n2: 2
5	<code>{ return x+y;}</code>	
6	<code>}</code>	Try 2:
7		-3 4
8	<code>namespace n2</code>	namespace n1: 1 namespace
9	<code>{</code>	n2: -7
10	<code>int function (int x,int y)</code>	
11	<code>{return x-y;}</code>	
12	<code>}</code>	
13		
14	<code>int main()</code>	
15	<code>{</code>	
16	<code>int a,b;</code>	
17	<code>cin&gt;&gt;a&gt;&gt;b;</code>	
18	<code>int x1=n1::function(a,b);</code>	
19	<code>//namespace n1</code>	
20	<code>cout&lt;&lt;"namespace:"&lt;&lt;x1&lt;&lt;endl;</code>	
21	<code>int x2=n2::function(a,b);</code>	
22	<code>//namespace n2</code>	
23	<code>cout&lt;&lt;"namespace n2:"&lt;&lt;x2&lt;&lt;endl;</code>	
24		
25	<code>return 0;</code>	
26	<code>}</code>	

### Тапсырма:

1. Екі есім кеңістігін жазыңыз: экономика және инженерия. Екеуінде де ауыспалы баға және экономика мен инженерияға сәйкесінше 500 және 700 қайтаратын есептеу функциясы бар.



## 59. Тақырып: Аргументтер тасымалдау

Функция аргументтерін тасымалдаудың үш түрі бар: мәндері арқылы, сілтемелер арқылы, мекен-жайлары арқылы.

1. Мәндер арқылы. C++ бағдарламасында аргументтер өздігінен мәндер арқылы тасымалданады. Аргументтер мәндер арқылы тасымалданған кезде функцияға көшірмесі жіберіледі.

2. Сілтемелер арқылы тасымалдау. Аргументтер сілтеме арқылы тасымалданғанда бағдарламашы аргумент мәндерін өзгерте алады. Сілтемелер арқылы тасымалданғанда мекен-жай белгісі (&) қолданылады. `void function(int &y) // y – бұл сілтеме айнымалысы.`

3. Мекен-жай арқылы тасымалдау. Мекен-жай арқылы тасымалдағанда айнымалы аргументі жіберілмейді, тек мекен-жай жіберіледі. Аргумент мекен-жай болып табылады. Функция параметрі pointer болу қажет.

`void function(int *y) // y – бұл сілтеуіш.`

`function(&a); // бағдарламашы main-ге сілтеуішті айнымалының мәніне емес, айнымалының өзіне жібереді.`

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	3 5
3	<code>void f1(int,int &amp;);</code>	3 5
4	<code>void f2(int,int *);</code>	3 5
5	<code>int main()</code>	6 10
6	<code>{</code>	3 10
7	<code>int a,b;</code>	-----
8	<code>cin&gt;&gt;a&gt;&gt;b;</code>	3 10
9	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;b&lt;&lt;endl;</code>	3 10
10	<code>f1(a,b);</code>	9 30
11	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;b&lt;&lt;endl;</code>	3 30
12	<code>cout&lt;&lt;"-----"&lt;&lt;endl;</code>	
13	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;b&lt;&lt;endl;</code>	
14	<code>f2(a,&amp;b);</code>	
15	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;b&lt;&lt;endl;</code>	
16	<code>return 0;</code>	
17	<code>}</code>	
18	<code>void f1(int a,int &amp;b){</code>	
19	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;b&lt;&lt;endl;</code>	
20	<code>a=a*2;</code>	
21	<code>b=b*2;</code>	
22	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;b&lt;&lt;endl;}</code>	
23	<code>void f2(int a,int *b){</code>	
24	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;*b&lt;&lt;endl;</code>	
25	<code>a=a*3;</code>	
26	<code>*b&gt;(*b)*3;</code>	
27	<code>cout&lt;&lt;a&lt;&lt;" "&lt;&lt;*b&lt;&lt;endl;</code>	

### Тапсырма:

1. Екі бүтін санның орнын сілтемемен, мекен-жаймен ауыстыратын екі функция жазыңыз.

## 60. Тақырып: Класс

Класс – мәліметтер құрылымының кеңейтілген концепциясы: тек мәліметтерді ғана ұстап тұрмай, сонымен қатар функцияларды да сақтай алады. Кластың экземпляр объектісі. Айнымалылар терминінде класс типі болады, ал объект айнымалы болады. Құрылымы:

```
class класс_аты {  
    айнымалылар және функциялар  
};
```

	C++ коды	Нәтиже
1	#include<iostream>	Try 1: Arman 25 Name: Arman Age: 25
2	#include<string>	
3	using namespace std;	Try 2: Andrey 19 Name: Andrey Age:19
4	class person //class description	
5	{	
6	public:	
7	string name; //variable	
8	int age; //variable	
9	void print() //function	
10	{	
11	cout<<"Name:"<<name<<endl;	
12	cout<<"Age:"<<age<<endl;	
13	}	
14	};//here it is important to put';' sign	
15		
16	int main()	
17	{	
18	person student; //object of class person	
19	string n;	
20	int a;	
21	cin>>n>>a;	
22	student.name=n;	
23	student.age=a;	
24	student.print();	
25	return 0;	
26	}	

### Тапсырма:

1. 3 айнымалысы бар дата класын жазыңыз: жыл, ай және күн. Датаны “күні айы жылы” форматында шығаратын print функциясын жазыңыз.

2. 2 айнымалысы бар кітап класын жазыңыз: кітап аты, авторы. Кітаптарды “автор аты” форматында шығаратын show функциясын жазыңыз.

## 61. Тақырып: Рұқсат спецификаторы

Рұқсат спецификаторы кластағы мәліметтерге рұқсат беру үшін декларация болып табылады. Спецификатор рұқсаты келесі үш кілтсөздің бірі: private, public немесе protected.

- public – мүшелері көрінетін барлық жерден қолжетімді;
- private – мүшелері өзі жазылған класта немесе достарына ғана қолжетімді;
- protected – мүшелері сол кластағы басқа мүшелерден немесе достарынан қолжетімді. Сонымен қатар оның туынды кластарынан (бала кластарынан) да.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1: Askar 25
2	#include<string>	Name: Askar
3	using namespace std;	Age: 25
4	class person{	
5	private://variable name and age are private	Try 2: Arman 33
6	string name;	Name: Arman
7	int age;	Age:33
8	public: //function are public	
9	void set(string a,int b){	
10	name=a;	
11	age=b;}	
12	void print(){	
13	cout<<"Name:"<<name<<endl;	
14	cout<<"Age:"<<age<<endl;}	
15	};	
16		
17	int main(){	
18	person student;	
19	string n;	
20	int a;	
21	cin>>n>>a;	
22	student.set(n,a);//it works because of public	
23		
24	//student.name=n;//without comment it is CE	
25	//student.age=a;//because they are private	
26		
27	student.print();//again public	
28	return 0;	
29	}	

### Тапсырма:

1. Екі setInfo және getInfo атты public функциялары мен 3 private айнымалы аттары, GPA, курсы бар student класын жазыңыз.

## 62. Тақырып: Қолжетімділік функциясы

Класса функция декларациясының прототипі болуы мүмкін, бірақ оның іске асырылуы кластың сыртында болуы мүмкін. Кластың сыртында болған жағдайда, бағдарламалаушы функция осы кластың мүшесі екеніне және ол әдеттегі ауқымды функция еместігіне көз жеткізу үшін (::) орта операторын қолдануы тиіс. Класста бағдарламалаушылар кластың сыртында орналасқан класс мүшелерін белгілеу үшін (::, екі кос нүкте) орта операторын қолданады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>#include&lt;string&gt;</code>	Askar 3 3.8
3	<code>using namespace std;</code>	Name:Askar
4	<code>class student{</code>	Course:3
5	<code>string name;</code>	GPA:3.8
6	<code>double GPA;</code>	-----
7	<code>int course;</code>	Andrey 1 2.5
8	<code>public:</code>	Name: Andrey
9	<code>void set(string,int,double);// only prototype</code>	Course:1
10	<code>void print(){</code>	GPA:2.5
11	<code>cout&lt;&lt;"Name:"&lt;&lt;name&lt;&lt;endl;cout&lt;&lt;"Course:"&lt;&lt;</code>	
12	<code>course&lt;&lt;endl;cout&lt;&lt;"GPA:"&lt;&lt;GPA&lt;&lt;endl;}</code>	
13	<code>};</code>	
14	<code>void student::set(string a,int b,double c)//implementation</code>	
15	<code>{name=a;</code>	
16	<code>course=b;</code>	
17	<code>GPA=c;}</code>	
18		
19	<code>int main()</code>	
20	<code>{</code>	
21	<code>student s1,s2;</code>	
22	<code>string n;</code>	
23	<code>int a;</code>	
24	<code>double d;</code>	
25	<code>cin&gt;&gt;n&gt;&gt;a&gt;&gt;d;</code>	
26	<code>s1.set(n,a,d);</code>	
27	<code>s1.print();</code>	
28	<code>cout&lt;&lt;"-----"&lt;&lt;endl;</code>	
29	<code>cin&gt;&gt;n&gt;&gt;a&gt;&gt;d;</code>	
30	<code>s2.set(n,a,d);</code>	
31	<code>s2.print();</code>	
32	<code>return 0;}</code>	

### Тапсырма:

1. Ұзындығы және ені деген private айнымалысы бар, setWidth (енін орнату), setLength (ұзындығын орнату) және Area (ауданы) атты функциялары бар Rectangle класын жазыңыз. Функцияның барлық жүзеге асырылуларын кластың сыртына жазыңыз.

### 63. Тақырып: Конструктор

Конструктор – бұл кластағы айнымалыларды инициалдау үшін қолданылатын арнайы функция. Ол осы кластың жаңа объектісі құрылғанына, құрылмағанына қарамастан автоматты түрде шақырылады. Бұл конструктор функциясының аты кластың атымен бірдей болуы керек және оның ешқандай қайтару типі, тіпті void типі болмау керек. Құрылымы:

Класс\_аты(мәліметтер\_типі параметр)

{ C++ коды }

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	#include<string>	Askar 2 3.5
3	using namespace std;	New student created
4	class student{	Name:Askar
5	string name;	Course:2
6	double GPA;	GPA:3.5
7	int course;	-----
8	public:	Olzhas 1 3.8
9	student(string,int,double);//constructor	New student created
10	void print()	Name: Olzhas
11	};	Course:1
12	void student::print(){	GPA:3.8
13	cout<<"Name:"<<name<<endl;	
14	cout<<"Course:"<<course<<endl;	
15	cout<<"GPA:"<<GPA<<endl;}	
16	student::student(string a,int b,double	
17	c)//implementation	
18	{	
19	cout<<"New student is created"<<endl;	
20	name=a;course=b;GPA=c;}	
21		
22	int main(){	
23	string n;	
24	int a;	
25	double d;	
26	cin>>n>>a>>d;	
27	student s1(n,a,d);	
28	s1.print();	
	return 0;}	

#### Тапсырма:

1. (int hour, int minute, int seconds) (сағат, минут, секунд) конструкторы бар time1 класын жазыңыз.

2. (string model, int year, string company) (моделі, жылы, компаниясы) конструкторы бар notebook класын жазыңыз.

## 64. Тақырып: Конструктордың артуы

Класта параметрлерінің түрлі типі мен саны бар бір немесе бірнеше конструкторлар болуы мүмкін. Бұл конструктордың артуы деп аталады. Олардың барлығының аты бірдей болу керек, бірақ компилятор параметрлері аргументтерімен сәйкес келетін конструкторларды ғана шақыратын болады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	3 5
3	class rectangle{	Without any parameter
4	int width,height;	Square is created
5	public:	Rectangle is created
6	rectangle();//with no parameters	9 15
7	rectangle(int);//with parameter width(square)	
8	rectangle(int,int);//with parameters width and height	Try 2:
9	int area();	4 7
10	};	Without any parameter
11	rectangle::rectangle(){	Square is created
12	width=height=0;	Rectangle is created
13	cout<<"Without any parameter"<<endl;}	16 28
14	rectangle::rectangle(int w){	
15	width=w;	
16	height=h;	
17	cout<<"Square is created"<<endl;}	
18	rectangle::rectangle(int w,int h){	
19	width=w;	
20	height=h;	
21	cout<<"Rectangle is created"<<endl;}	
22	int rectangle::area(){	
23	return width*height;}	
24		
25	int main{	
26	int a,b;	
27	cin>>a>>b;	
28	rectangle r1;//constructor 1	
29	rectangle r2(a);//constructor 2	
30	rectangle r3(b);//constructor 3	
31		
32	cout<<r2.area()<<" "<<r3.area();	
33	return 0;}	

### Тапсырма:

1. manufacture (зауыт), model (моделі), price (бағасы) және color (түсі) айнымалылары бар cellPhone класын жазыңыз. Осы класс үшін 2 арттырылған конструктор жазыңыз. changePrice (бағаны өзгерту) және changeColor (түсін өзгерту) функцияларын жазыңыз.

2. name (аты), author (авторы), pages (беттері) және ISBN айнымалылары бар book класын жазыңыз. Осы класс үшін 3 арттырылған конструктор жазыңыз. changeName (атын өзгерту) және print (экранның шығару) функцияларын жазыңыз.

## 65. Тақырып: Стандартты конструктор

Егер класс анықтамасында конструктор декларациясы болмаса, компилятор класта аргументсіз стандартты конструктор бар деп есептейді. Бағдарламалаушы аргументсіз класс объектілерін жариялай береді. Бірақ бағдарламалаушы класқа арналған өз конструкторын жариялағандықтан, компилятор implicit стандартты конструкторын қамтамасыз етпейді. Сондықтан бағдарламалаушы кластың барлық объектілерін конструктор прототипіне сәйкес жариялағаны жөн.

<pre>class example 1 { public: int a; void seta(int b){ a=b;} };</pre>	<pre>class example 2 { public: int a; example2(int b) {a=b;} };</pre>
without constructor,so we can write like example 1 d; and cannot write like example1 d(5);	with constructor, so we can write like example 2 d(5); and cannot write like example2 d;

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	3
3	class example {	3 3
4	public:	
5	int a;	Try 2:
6	void seta (int b) {a=b;}	4
7	};	4 4
8	class example2{	
9	public:	
10	int a;	
11	example2 (int b) {a=b;}	
12	};	
13	int main()	
14	{	
15	int x;	
16	cin>>x;example1 r1;r1.setA(x);	
17	example2 r2(x);	
18	cout<<r1.a<<" "<<r2.a;	
19	return 0;	
20	}	

### Тапсырма:

1. print (шығару) функциясымен конструкторы бар «инженерлік топ» және конструкторы жоқ «экономикалық топ» класын жазыңыз. Оларды қолдану үшін main class жазыңыз.

## 66. Тақырып: Деструктор

Деструктор дегеніміз – конструктордың қарама-қарсысы. Әдетте ол объект жойылған кезде автоматты түрде шақырылады, себебі оның әрекет ету ортасы аяқталды немесе объект delete операторын қолдана отырып, динамикалық түрде теңестіріліп босатылды. Немесе өзінің жеке деструкторын жасаса болады.

Құрылымы:

```
класс_аты () {  
    “ с++ коды  
}
```

Ескерту: класта тек бір ғана деструктор бола алады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	#include<string>	3 5
3	using namespace std;	3 5
4	class book{	3 5
5	string*name;	6 10
6	int*ISBN;	3 10
7	public:	-----
8	book(string n,int i); //constructor	3 10
9	~book(); //destructor	3 10
10	};	9 30
11	book::book(string n,int i){	3 30
12	name=new string(n);ISBN=new int;	
13	*ISBN=i;	
14	cout<<"object"<<n<<endl;}	
15	book::~~book(){	
16	cout<<"Object is destructed"<<endl;	
17	delete ISBN;	
18	delete name;}	
19		
20	int main(){	
21	string s;	
22	int x;	
23	cin>>s>>x;	
24	{book b(s,x); }	
25	//here destructor is working book b2("AAA",x);	
26	return 0;	
27	//here destructor is working	

### Тапсырма:

1. Деструкторы бар student класын жазыңыз.



## 67. Тақырып: Сыртқы класс

Бағдарламалаушы басқа ресурстарды қолдану үшін класты белгілеп іске асыра алады.

Көбінесе (.h) бастапқы файлда кластың анықтамасы болады. Және (.cpp)-де осы кластың жүзеге асырылуы бар.

C++ CODE		
	example.h	example.cpp
1	class example	#include "example.h"
2	{	#include<iostream>
3	int i;	using namespace std;
4	double d;	
5	public:	example::example(){ }
6	example();	
7	void set(int,double);	void example::set(int a,double q){
8	void print();	i=a;
9	};	d=q;}
10		
11		void example::print(){
12		cout<<"Integer"<<i<<endl;
13		cout<<"Double"<<d<<endl;}
	main.cpp	RESULT
1	#include <iostream>	Try 1:
2	#include "example.h"	2 3.1
3	using namespace std;	Integer 3
4		Double 3.1
5	int main()	
6	{	Try 2:
7	int x;	4 4.11
8	double d;	Integer 4
9	cin>>x>>d;	Double 4.11
10	example s1;	
11	s1.set(x,d);	
12	s1.print();	
13	return 0;}	

### Тапсырма:

1. Бастапқы және анықталған класс арқылы book C++ класын жазыңыз. Тестілеу үшін main класын жазыңыз.

### 68. Тақырып: Класқа нұсқау

C++-та бағдарламалаушы класқа нұсқау жасай алады. Құрылымы:

класс\_аты \*объект\_аты;

Функцияны шақырудың екі тәсілі бар:

1 - (\*объект\_аты).method();

2 — объект\_аты >method();

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	#include<string>	Name Unknown
3	using namespace std;	Author Unknown
4	class Book{	Abay
5	string name,author;	Kozimnin karasi
6	public:	Name Abay joli
7	Book();	Author Auezov
8	void setBook(string,string);	Name Abay
9	void print();	Author Kozimnin
10	};	
11	Book::Book{	Try 2:
12	name="Unknown";	Name Unknown
13	author="Unknown";}	Author Unknown
14	void Book::setBook(string n,string a){	Okulik
15	name=n;	Altynsarin
16	author=a;}	Name Abay joli
17	void Book::print(){	Author Auezov
18	cout<<"Name"<<name<<endl;	Name Okulik
19	cout<<"Author"<<author<<endl;{	Author Altynsarin
20		
21	int main()	
22	{	
23	Book r1;	
24	r1.print();	
25	string a,b;	
26	cin>>a>>b;	
27	Book *r2=new Book[2];	
28	r2->setBook("Abay joli","Auezov");	
29	r2[0].print();	
30	r2[1].setBook(a,b);	
31	r2[2].print();	
32		
33	return 0;}	

#### Тапсырма:

1. Класқа нұсқауы бар mobile phone атты C++ класын жазыңыз.

## 69. Тақырып: “this” нұсқаулығы

Осы сөз нұсқаулықтағы арнайы тип болып табылады. Бұл айтылған тақырыптың мекен-жайын сақтау үшін қолданылады. Сіз А класына x атты тақырып дайындадыңыз делік, және А класында f() статикалық емес мүшесі бар функция бар. Егер сіз функцияны x.f() деп атасаңыз, осы құпия сөз x мекен-жайын сақтайды. Сіз оған одан артық өзгерістер енгізе алмайсыз.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	2
3	class calc{	2
4	int a;	0
5	public:	15
6	calc();	
7	void set(int);	Try 2:
8	int get();	4
9	calc&plus(int);	4
10	};	0
11	calc::calc(){	15
12	a=0;}	
13	void calc::set(int x){	
14	a=x;}	
15	int calc::get(){	
16	return a;}	
17	calc&calc::plus(int x){	
18	a+=x;	
19	return*this;}	
20		
21	int main(){	
22	calc c1;	
23	calc c2;	
24	int x;	
25	cin>>x;	
26	c1.set(x);	
27	cout<<c1.get()<<endl;	
28	cout<<c2.get()<<endl;	
29	c2.plus(10).plus(5);	
30	cout<<c2.get()<<endl;	
31	return 0;}	

### Тапсырма:

1. C++-та осы нұсқаулық және тақырып арқылы кластың нөмірін жазыңыз. Нұсқаулық тақырыппен сәйкес келетін тексеру функциясын енгізіңіз.

## 70. Тақырып: Қайта іске қосу операторлары

Класта қолдануға операторларды арттыру үшін бағдарлама жасаушы операторлық функцияны анықтауы керек, олар әрине тұрақты функция болуы керек, және олардың аттары оператор құпия сөзі мен оператор белгісімен сәйкес келуі керек. Құрылымы мынадай:

мәлімет\_типi оператор белгісі (параметрлері)

{ C++ коды }

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	1 2
3		3 3
4	class number{	
5	public:	Try 2:
6	int x;	3 7
7	number() {};	10 10
8	number(int);	
9	number operator+(number);	
10	};	
11	number::number(int a){	
12	x=a;}	
13	number::number operator+(number n1){	
14	number temp;	
15	temp.x=x+n1.x;	
16	return temp;	
17	//return number(x+n1.x);	
18	}	
19		
20	int main(){	
21	int a1,a2;	
22	cin>>a1>>a2;	
23	number a(a1);	
24	number b(a2);	
25	number c,d;	
26	c=a+b;;	
27	d=a.operator+(b);	
28	cout<<c.x<<" "<<d.x<<endl;	
29	return 0;}	

### Тапсырма:

1. Екі нөмірді анықтайтын number класын жазыңыз.

## 71. Тақырып: Унарлы операторды арттыру

Унарлы оператор қолданылса, ол бір ғана операндпен жұмыс істейді, осыған сәйкес операнд шақырушы болады және ешқандай дәлел қолданылмайды.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	1 2
3		2 3
4	<code>class number{</code>	
5	<code>public:</code>	Try 2:
6	<code>int x;</code>	34 45
7	<code>number() {};</code>	35 46
8	<code>number(int);</code>	
9	<code>number operator++();</code>	
10	<code>number operator ++(int);</code>	
11	<code>};</code>	
12	<code>number::number(int a){</code>	
13	<code>x=a;}</code>	
14	<code>number number::operator++()</code>	
15	<code>{return number(++x);}</code>	
16	<code>number number::operator++(int);</code>	
17	<code>{return number(x++);}</code>	
18		
19	<code>int main(){</code>	
20	<code>int a1,a2;</code>	
21	<code>cin&gt;&gt;a1&gt;&gt;a2;</code>	
22	<code>number a(a1);</code>	
23	<code>number b(a2);</code>	
24	<code>a++;</code>	
25	<code>++b;</code>	
26	<code>cout&lt;&lt;a.x&lt;&lt;" "&lt;&lt;b.x&lt;&lt;endl;</code>	
27	<code>return 0;}</code>	

### Тапсырма:

1. Декремент операторымен number класын жазыңыз.

## 72. Тақырып: Бинарлы операторды арттыру

Бинарлы оператор қолданылғанда, ол екі операндпен жұмыс істейді, осыған сәйкес бірінші операнд шақырушы ал екінші операнд дәлел ретінде қолданылады. Ол бинарлы операторларды арттыруда бір ғана дәлел қабылдаудағы мәжбүрліктен көрініс табады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	2 3
3		Not equal
4	<code>class number{</code>	
5	<code>public:</code>	Try 2:
6	<code>int x;</code>	3 3
7	<code>number() {};</code>	Equal
8	<code>number(int);</code>	
9	<code>bool isEqual(number);</code>	
10	<code>};</code>	
11	<code>number::number(int a){</code>	
12	<code>x=a;}</code>	
13	<code>bool number::operator==(number n1);</code>	
14	<code>return(x==n1.x);}</code>	
15		
16	<code>int main(){</code>	
17	<code>int a1,a2;</code>	
18	<code>cin&gt;&gt;a1&gt;&gt;a2;</code>	
19	<code>number a(a1);</code>	
20	<code>number b(a2);</code>	
21	<code>if(a==b) cout&lt;&lt;"Equal"&lt;&lt;endl;</code>	
22	<code>else cout&lt;&lt;"Not equal ";</code>	
23	<code>return 0;</code>	
24	<code>}</code>	

### Тапсырма:

1. Сандардың үлкен, кіші немесе тең екендігін тексеретін number класын жазыңыз.

### 73. Тақырып: Енгізу/шығару операциясын арттыру

C++-та cin және cout функциялары бар. Бұл дегеніміз бағдарлама жасаушы мәліметті өзіне түсінікті етіп класс жасауына болады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	3 5
3		x:3 y:5
4	class Cnumber{	4 5
5	public:	x:4 y:5
6	int x,y;	
7	Cnumber() {};	Try 2:
8	Cnumber(int,int);	5 5
9	friend	x:5 y:5
10	istream&operator>>(istream%in,Cnumber &t);	12 13
11	friend	x:12 y:13
12	ostream&operator>>(ostream&out,Cnumber	
13	&t);	
14	};	
15	Cnumber::Cnumber(int a,int b){	
16	x=a;	
17	y=b;}	
18	istream&operator>>(istream&in,Cnumber &t){	
19	in>>t.x;in>>t.y;	
20	return in;}	
21	ostream&operator<<(ostream &out,Cnumber	
22	&t){	
23	out<<"x:"<<t.x<<"y:"<<t.y<<endl;	
24	return out;}	
25		
26	int main(){	
27	Cnumber a1,a2,a3;	
28	cin>>a1;	
29	cout<<a1;	
	cin>>a2;	
	cout<<a2;	
	return 0;}	

#### Тапсырма:

1. Енгізу/шығару операторын арттыру арқылы number класын жазыңыз.
2. int hour (сағат), int minute (минут), int seconds (секунд) айнымалылары бар time1 класын жазыңыз. Енгізу/шығару операторын арттыру.

#### 74. Тақырып: Тату функция

Тату функция сол кластың мүшесі болмаса да, жеке кластың мүшесі екенін анықтайтын функция болып табылады. Ал басқа жағынан алатын болсақ, ол жай ғана функция болып табылады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	3 5
3		8
4	<code>class number{</code>	
5	<code>public:</code>	Try 2:
6	<code>int x;</code>	4 7
7	<code>number() {};</code>	11
8	<code>number(int);</code>	
9	<code>friend number operator+(number,number);</code>	
10	<code>void print(){cout&lt;&lt;x;}</code>	
11	<code>};</code>	
12	<code>number::number(int a){</code>	
13	<code>x=a;}</code>	
14	<code>number operator+(number t1,number t2){</code>	
15	<code>number temp;</code>	
16	<code>temp.x=t1.x+t2.x;</code>	
17	<code>return temp;}</code>	
18		
19	<code>int main(){</code>	
20	<code>int a1,a2;</code>	
21	<code>cin&gt;&gt;a1;</code>	
22	<code>number a(a1);</code>	
23	<code>cin&gt;&gt;a2;</code>	
24	<code>number b(a2);</code>	
25		
26	<code>number c;</code>	
27	<code>c=a+b;;</code>	
28	<code>c.print();</code>	
29	<code>return 0;}</code>	

#### Тапсырма:

1. Екі аргументі бар класс мүшелері үшін тату операторды жазыңыз: біріншісі объектпен, екіншісі бүтін санмен.

- Жүйемен
- Бүтін санмен



### 75. Тақырып: Тату класс

Тату класс деп бір-бірімен тату болып табылатын класты айтады. Бұл бірінші класқа екінші кластың protected және private мүшелеріне рұқсат береді.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	3
3		9
4	<code>class CSquare;</code>	
5	<code>class CRectangle{</code>	Try 2:
6	<code>int width,height;public:</code>	4
7	<code>int area() {return(width * height);};</code>	16
8	<code>void convert(CSquare a);</code>	
9	<code>};</code>	
10	<code>void CRectangle::convert (CSquare a){</code>	
11	<code>width=a.side;</code>	
12	<code>height=a.side;</code>	
13	<code>}</code>	
14		
15	<code>class CSquare{</code>	
16	<code>private:int side;public:</code>	
17	<code>void set_side(int a) {side=a;}</code>	
18	<code>friend class CRectangle;</code>	
19	<code>};</code>	
20		
21	<code>int main()</code>	
22	<code>{</code>	
23	<code>CSquare sqr;</code>	
24	<code>CRectangle rect;</code>	
25	<code>int a;</code>	
26	<code>cin&gt;&gt;a;</code>	
27	<code>sqr.set_side(a);</code>	
28	<code>rect.convert(sqr);</code>	
29	<code>cout&lt;&lt;rect.area();</code>	
30	<code>return 0;}</code>	

## 76. Тақырып: Мұрагерлік

Мұрагерлік дегеніміз басқа кластан тағы да кластар шығару жолы, және олар өздеріне тікелей барлық мүшелерін де қосып алады. Басқа класс бастапқы болып табылады, екінші класс бағынышты болып табылады.

	C++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	Askar Askarov
3	<code>class person{</code>	2.3
4	<code>public:</code>	1
5	<code>string name;</code>	Student name:Askar
6	<code>int age;</code>	GPA:2.3 Course:1
7	<code>void setName(string s) {name=s;}</code>	
8	<code>void setAge(int i) {age=i;}</code>	Try 2:
9	<code>string getName() {return name;}</code>	Arman Amangali
10	<code>int getAge(){return age;}</code>	3.6
11	<code>void print(){</code>	3
12	<code>cout&lt;&lt;"name:"&lt;&lt;name&lt;&lt;"age:"&lt;&lt;age&lt;&lt;endl;}</code>	Student name:Arman
13	<code>};</code>	GPA:3.6 Course:3
14	<code>class student:public person{</code>	
15	<code>public;</code>	
16	<code>double GPA;int course;</code>	
17	<code>void setGPA(double d){GPA=d;}</code>	
18	<code>void setCourse(int i){course=i;}</code>	
19	<code>void info(){</code>	
20	<code>cout&lt;&lt;"Student name:"&lt;&lt;name&lt;&lt;endl;</code>	
21	<code>//cout&lt;&lt;"Student surname:"&lt;&lt;surname&lt;&lt;endl;</code>	
22	<code>//line up cannot be used because surname is private;</code>	
23	<code>cout&lt;&lt;"GPA:"&lt;&lt;GPA&lt;&lt;"Course:"&lt;&lt;course;}</code>	
24	<code>};</code>	
25	<code>int main(){</code>	
26	<code>student s1,string name; surname;</code>	
27	<code>cin&gt;&gt;name;</code>	
28	<code>s1.setName(name);</code>	
29	<code>cin&gt;&gt;surname;</code>	
30	<code>s1.setSurname(surname);</code>	
31	<code>double g;int c;</code>	
32	<code>cin&gt;&gt;g&gt;&gt;c;</code>	
33	<code>s1.setGPA(g);</code>	
34	<code>s1.setCourse(c);</code>	
35	<code>s1.info();</code>	
36	<code>return 0;}</code>	

### Тапсырма:

1. Қорғалған `int` `x`, `u`-пен `point` (нүкте) класын жазыңыз. `Line` (сызық) мұрагер класын жазыңыз.

## 77. Тақырып: Қорғалған мүше

Қорғалған деңгей жеке деңгейге ұқсас. Өзгешелік тек қана мұрагерліктен көрінеді. Егер де бір класс басқа кластан құрылса, барлық мүшелері бастапқы класпен қорғалады, тек жеке мүшелері емес.

	С++ коды	Нәтиже
1	<code>#include&lt;iostream&gt;</code>	Try 1:
2	<code>using namespace std;</code>	Askar
3	<code>class person{</code>	21
4	<code>public:</code>	Name:Askar age:21
5	<code>string surname;</code>	3.4
6	<code>protected:string name; int age;</code>	2
7	<code>public:</code>	Name:Askar age:21
8	<code>void setName(string s) {name=s;}</code>	GPA:3.4 Course:2
9	<code>void setAge(int i) {age=i;}</code>	
10	<code>void setSurname(string s) {surname=s;}</code>	Try 2:
11	<code>void print(){</code>	Arman
12	<code>cout&lt;&lt;"name:"&lt;&lt;name&lt;&lt;"age:"&lt;&lt;age&lt;&lt;endl;}</code>	19
13	<code>};</code>	Name:Arman age:19
14	<code>class student:public person{</code>	2.6
15	<code>public;</code>	1
16	<code>double GPA;int course;</code>	Name:Arman age:19
17	<code>void setGPA(double d){GPA=d;}</code>	GPA:2.6 Course:1
18	<code>void setCourse(int i){course=i;}</code>	
19	<code>void info(){</code>	
20	<code>print();</code>	
21	<code>cout&lt;&lt;"GPA:"&lt;&lt;GPA&lt;&lt;"Course:"&lt;&lt;course;}</code>	
22	<code>};</code>	
23	<code>int main(){</code>	
24	<code>student s1,name; int age;</code>	
25	<code>cin&gt;&gt;name&gt;&gt;age;</code>	
26	<code>s1.setName(name);</code>	
27	<code>s1.Age(age);</code>	
28	<code>s1.print();</code>	
29	<code>double g;int c;</code>	
30	<code>cin&gt;&gt;g&gt;&gt;c;</code>	
31	<code>s1.setGPA(g);</code>	
32	<code>s1.setCourse(c);</code>	
33	<code>s1.info();</code>	
34	<code>return 0;}</code>	

### Тапсырма:

1. width (ені) және height (биіктігі) айнымалылары бар shape (өлшем) класын жазыңыз. Area (аудан) функциясы бар rectangle (тік төртбұрыш) мұрагер класын жазыңыз.

## 78. Тақырып: Мұрагерлік байланыс

Мұрагерлік байланыстың үш түрлі жолы бар: жаопы, жеке және қорғалған.

class Child: public Parent { }; Барлық мүшелері бастапқы қолжетімділік деңгейін сақтайды. Private мүшелер private болып, protected мүшелер protected болып, және public мүшелер public болып қалады.

class Child: private Parent { }; private мүшелер private болып қала береді, ал protected пен public мүшелер private болады.

class Child: protected Parent { }; public пен protected мүшелер protected болады, ал private мүшелер private болып қала береді.

class Child: Parent { }; үнсіздік бойынша private қолданылады.

	C++ коды	Нәтиже
1	#include<iostream>	Try 1: Compilation error  Class square tries to use shape's width and height which are private for it.
2	using namespace std;	
3	class shape{	
4	private:	
5	int Z;	
6	protected:	
7	int width,height;	
8	public:	
9	void set(int a, int b){	
10	width=a;	
11	height;}	
12	};	
13	class rectangle:private shape{	
14	public:	
15	int getArea(){return width*height;}	
16	};	
17	class square:public rectangle{	
18	public:	
19	int getArea() {return width*height;}	
20	};	
21		
22	int main(){	
23	rectangle r;	
24	square s;	
25	return 0;}	
26		
27		

### Тапсырма:

1. Берілген кодты дұрыс жұмыс істейтіндей етіп өзгертіңіз.

## 79. Тақырып: Алынған кластың инициализациясы

Ережеге сәйкес, бағынышты класс бастапқы кластың барлық мүшелерін қабылдайды, мыналардан басқасын:

- оның құрастырушысы мен бұзушысы
- оның =() операторының мүшелері
- оның достары

Негізгі класс конструкторын қайта іске қосу үшін алынған жаңа объект келесі құрылым арқылы құрылады:

алынған\_конструктор\_аты (параметрлері):

негізгі\_конструктор\_аты (параметрлері)

{ C++ коды }

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	5
3	class A{	3.4
4	public:	Z
5	A(int n) {cout<<"A:"<<n<<endl;}	A:5
6	};	B:3.4
7	class B: public A	C:Z
8	{	
9	public:	Try 2:
10	B(int n,double d): A(n){	9
11	cout<<"B:"<<d<<endl;}	4.1q
12	};	A:9
13	class C: public B{	B:4.1
14	public:	C:q
15	C(int n,double d,char ch):B(n,d){	
16	cout<<"C:"<<ch<<endl;}	
17	};	
18		
19	int main()	
20	{	
21	int i1;	
22	double i2;	
23	char i3;	
24	cin>>i1>>i2>>i3;	
25	C c1(i1,i2,i3);	
26	return 0;	
27	}	

## 80. Тақырып: Көптік мұрагерлік

Көптік мұрагерлік алынған класты көп ата-анадан (parent) мүшелерін мұрагерленуге мүмкіндік береді. Құрылымы мынадай:

```
class алынған_класс_аты : public parent1_класс, public parent2_класс { C++ коды };
```

	C++ коды	Нәтиже
1	#include<iostream>	Try 1:
2	using namespace std;	Arman
3	class person{	8 701 9999999
4	protected:	2
5	string name;	Student name:Araman
6	public:	Phone number:87019999999
7	void setName(string n) {name=n;}	Course:2
8	};	
9	class mobilePhone{	Try 2:
10	protected:	Askar
11	string phone;	2738800
12	public:	1
13	void setPhone(string n){phone=n;}	Student name:Askar
14	};	Phone number:2738800
15	class student:public person,public	Course:1
16	mobilePhone{	
17	int course;	
18	public:	
19	void info(){	
20	cout<<"Student name:"<<name<<endl;	
21	cout<<" Phone number:"<<phone<<endl;	
22	cout<<"Course:"<<course<<endl;}	
23	void setCourse(int i){course=i;}	
24	};	
25	int main(){	
26	string n,p;	
27	student s1;	
28	cin>>name;	
29	s1.setName(name);	
30	cin>>p;	
31	s1.setPhone(p);	
32	int i;	
33	cin>>i;	
34	s1.setCourse(i);	
35	s1.info();	
	return 0;}	

Тапсырма:

1. father және mother атты 2 ата-ана класы бар child класын жазыңыз.

## Сілтемелер

1. [www.cplusplus.com](http://www.cplusplus.com)
2. [www.learncpp.com](http://www.learncpp.com)
3. <http://www.codingunit.com>
4. <http://www.mochima.com/tutorials>