

# Лекція 2. Програмування

*Система типів. Літерали*

*Математичні функції*

*Використання cin, cout*

# Змінна. Тип даних

- *Змінна* (програми) – іменована ділянка пам'яті комп'ютера, в яку можна поміщати значення, і звідки можна це значення отримувати
- У типізованих мовах програмування *кожна* змінна зберігає дані деякого заданого *типу*
- C++ – строго типізована, статично типізована
- *Типом* даних у програмуванні називають множину допустимих значень, для яких фіксовано спосіб кодування до двійкового вигляду і визначено перелік допустимих операцій
- Спосіб кодування пов'язано з розміром пам'яті, виділеної для одного значення

# Система типів мови C++

- Вбудовані типи
  - Фундаментальні
    - Арифметичні
      - Інтегральні
        - Логічний – *bool*
        - Символьні – *char*, *wchar\_t*
        - Цілі – *[unsigned] short*, ***int***, *long*, *long long*
      - Дійсні – *float*, ***double***, *long double*
    - Тип *void* – відсутні дані
  - Вказівники, наприклад *int\**
  - Масиви, наприклад *char[]*
  - Посилання, наприклад *int&*
- Оголошені користувачем
  - Переліки – *enum*
  - Структури – *struct*
  - Класи – *class*
  - Бітові поля
  - Об'єднання

*Вбудованим називають тип даних, інформацією про який (назва, кодування, операції) володіє компілятор*

# Логічний тип *bool*

- Містить лише два значення:
  - *false* – хиба, код 0;
  - *true* – істина, код 1;
  - У пам'яті займає 1 байт: 0000 0000 == *false*; *true*, якщо хоча б один біт != 0
- Операції:
  - логічні: ! (заперечення), && (кон'юнкція), || (диз'юнкція);
  - порівняння: <, >, >=, <=, == (рівне), != (не рівне); *false* < *true*;
  - виведення в потік <<, введення з потоку >> (вводити потрібно числа: нуль позначає хибу, не нуль – істину);
  - присвоєння =, наприклад: *bool* a = *true*;  
*bool* b, c; b = *false*; c = a && b;
  - арифметичні (бо логічний є різновидом цілого), але робити цього не варто (*true* + *true* == *true*)

# *Таблиця істинності логічних операторів*

<b>a</b>	<b>b</b>	<b>! a</b>	<b>a &amp;&amp; b</b>	<b>a    b</b>
0	0	1	0	0
0	1	1	0	1
1	0	0	0	1
1	1	0	1	1

# Літерні типи *char*, *wchar\_t*

- *char* займає 1 байт, містить ASCII коди (діапазон від 0 до 255)
  - American Standard Code for Information Interchange
  - 'A' – велика літера а, 'a' – мала літера а, '5' – літера (цифра) п'ять, її код відрізняється від числа 5, '+' – літера плюс, ' ' – пропуск
  - спеціальні літери: '\n' – кінець рядка; '\r' – перевід каретки; '\f' – кінець сторінки; '\t' – горизонтальна табуляція (відступ на декілька літер); '\v' – вертикальна табуляція (відступ на декілька рядків); '\a' – звук; '\b' – повернення на одну літеру назад; '\\' – зворотна похила; '\"' – апостроф; '\"' – лапки; '\?' – знак питання; '\0' – літера з кодом нуль.
- *wchar\_t* займає 2 байти, містить коди UNICODE (від 0 до 65535)
  - UNlversal CODE
  - зображення літер L'A', L'5', L'є'
- *char16\_t*, *char32\_t*

# Логічні вирази

- число  $x$  – додатнє:
- $x \in [a; b]$ :
- літера  $c$  – мала латинська:
- літера  $c$  – голосна:
- ціле число  $k$  кратне 3:
- $x = \max(x, y, z)$ :
- $x \neq \max(x, y, z)$ :

$x > 0$  //  $\rightarrow$  true | false

$x \geq a \ \&\& \ x \leq b$

$'a' \leq c \ \&\& \ c \leq 'z'$

$c == 'a' \ || \ c == 'e' \ || \ c == 'i' \ ||$   
 $c == 'o' \ || \ c == 'u'$

$k \% 3 == 0$

$x > y \ \&\& \ x > z$

$!(x > y \ \&\& \ x > z)$

або  $x \leq y \ || \ x \leq z$

# Цілі типи

Назва	Розмір	Діапазон
<b>char; unsigned char</b>	1 байт	0..255
<b>signed char</b>	1 байт	−128..127
<b>short; short int; signed short</b>	2 байти	−32768..32767
<b>unsigned short; unsigned short int</b>	2 байти	0..65535
<b>int; signed int</b>	4 байти	−2147483648..2147483647
<b>unsigned int</b>	4 байти	0..4294967295 == $(2^{32}-1)$
<b>long або long int == int. unsigned long == unsigned int</b>	4 байти	−2147483648..2147483647 0..4294967295
<b>long long; long long int</b>	8 байтів	$-2^{63}..(2^{63}-1)$ ; $2^{63} \approx 9 \times 10^{18}$
<b>unsigned long long</b>	8 байтів	$0.. 2^{64} - 1$



# Зображення цілих

- звичні зі школи: 5; 2020; -29; +7 – всі типу `int`;
- з суфіксами, що вказують тип:
  - 5U; 17u – `unsigned`;
  - 23L – `long`;
  - 5987458ULL – `unsigned long long int`;
- з префіксами, що вказують основу числення:
  - 0b00001010 – двійковий,
  - 012 – вісімковий (префікс 0),
  - 0x0A – шістнадцятковий записи числа 10.

# Оператори над цілими

- арифметичні
  - бінарні  $+$ ,  $-$ ,  $*$  (множення),  $/$  (ділення),  $\%$  (обчислення остачі від ділення);
  - унарні  $+$ ,  $-$ ; префіксні та постфіксні  $++$ ,  $--$  (збільшення, зменшення на 1);
- порівняння  $<$ ,  $>$ ,  $>=$ ,  $<=$ ,  $==$ ,  $!=$ ;
- побітові:
  - $\sim$  (інверсія),
  - $\&$  (побітове і),  $|$  (побітове або),  $\wedge$  (додавання за модулем 2),
  - $<<$  (зсув ліворуч, аналог множення на 2),  $>>$  (зсув праворуч, аналог ділення на 2 без остачі);
- присвоєння:
  - звичайне (справа наліво)  $=$  ;
  - комбіновані (спочатку дія, потім присвоєння)  $+=$ ,  $-=$ ,  $*=$ ,  $/=$ ,  $\%=$ ,  $\&=$ ,  $|=$ ,  $\wedge=$ ,  $<<=$ ,  $>>=$  ;
- виведення в потік  $<<$ , введення з потоку  $>>$ .

# Приклади застосування операторів

- $2 / 3 == 0$
- $2 \% 3 == 2$ ;  $11 \% 3 == 2$
- `int x = 5; int y = -x; // y == -5`
- $0x13BA \& 0x00FF == 0x00BA$ 
  - 0001 0011 1011 1010  
0000 0000 1111 1111  
-----  
0000 0000 1011 1010
- `int x = 0b00110101; int y = ~x; // y = 0b11001010`
- $0b00001010 \ll 3 == 0b01010000$  //  $10 * 2^3 == 80$
- $a = b = c = 5;$

# Арифметичні операції:

- *трьохмісне додавання* (два операнди і результат)  **$c = a + b$** ; *// бінарний +*;
- *двохмісне додавання* (два операнди, результат потрапляє в лівий операнд)  
 $S = S + x$ ; мовою C++ таке додавання записують комбінованим присвоєнням  
 **$S += x$** ; *// «прочитати значення S, додати до нього x, результат записати в S»*;
- *одномісне додавання одиниці*, інкремент (збільшення операнда на 1)  
 $k = k + 1$ ; мовою C++ інкремент записують спеціальним оператором  **$++k$** ;  
*// префіксний інкремент*;
- *одномісне додавання одиниці «з пам'яттю»*, постфіксний інкремент  **$k++$** ;  
діє не так, як попередній, оскільки зберігає попереднє значення операнда;  
 $k++$  еквівалентне виконанню групи ( $t = k, k += 1, t$ ):
  - **`int i = 5; int j = ++i;`** *// i == 6, j == 6*
  - **`int i = 5; int j = i++;`** *// i == 6, j == 5*

# Дійсні типи

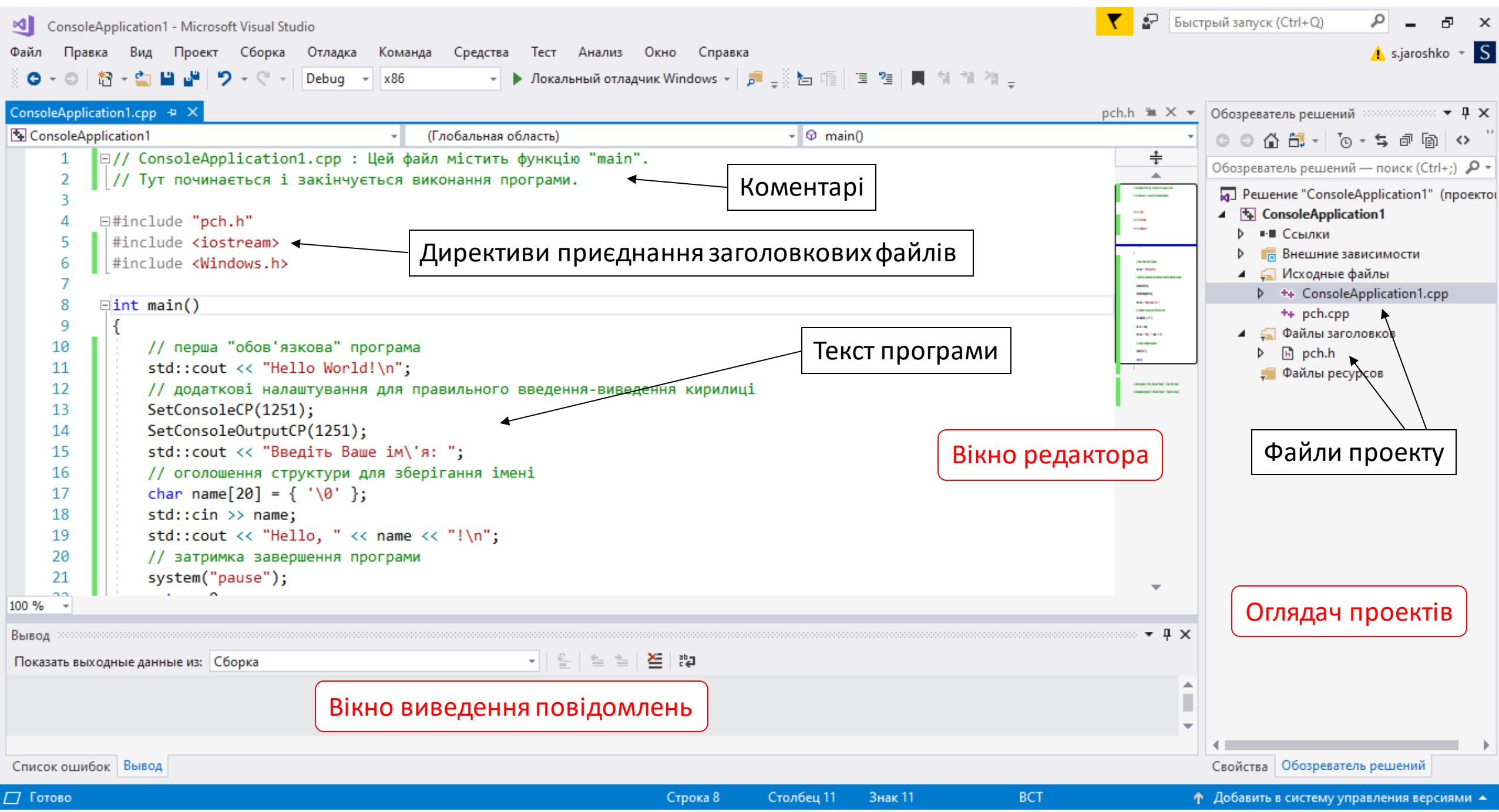
Назва	Розмір	Діапазон	Кількість цифр мантиси
<b>float</b>	4 байти	$\pm 3,4 \times 10^{-38} \dots \pm 3,4 \times 10^{38}$	до 8
<b>double</b>	8 байтів	$\pm 1,7 \times 10^{-308} \dots \pm 1,7 \times 10^{308}$	15
<b>long double</b>	8 байтів	те саме, що double	

- Зображення дійсних чисел:
  - $\pi \approx 3.14159265$  // **double**
  - $10^{-6} = 1e-6$  або  $1E-6$
  - 0.5 можна записати .5
  - дійсна одиниця 1.0 або 1.; від'ємне дійсне -1.25
  - Для вказання типу використовують суфікс F або L

# Оператори над дійсними

- арифметичні
  - бінарні +, −, \*(множення), /(ділення);
  - унарні +, −; префіксні та постфіксні ++, -- (збільшення, зменшення на 1);
- порівняння <, >, >=, <=, ==, !=;
- присвоєння:
  - звичайне (справа наліво) = ;
  - комбіновані (спочатку дія, потім присвоєння) +=, -=, \*=, /=, %=, &=, |=, ^=, <<=, >>= ;
- виведення в потік <<, введення з потоку >>.

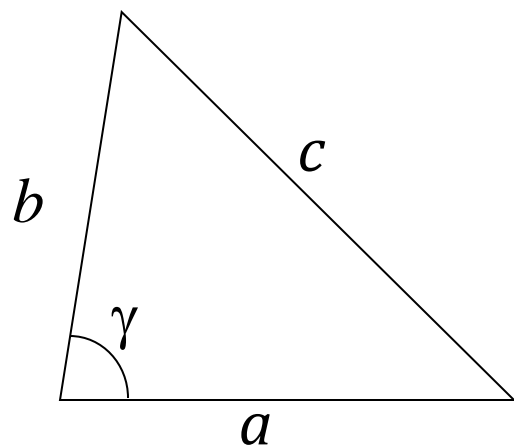
Пріоритет	Оператор	Асоціативність	Значення
1	::		Визначення діапазону доступу
	(вираз)		Групування
2	()	Л-П	Виклик функції
	[]		Індексування елементів масиву
	. ->		Прямий, опосередкований доступ до члена типу
	++ --		Постфіксні унарні інкремент/декремент
3	! ~	П-Л	Заперечення, інверсія
	+ -		Унарні (перед змінною, перед дужкою)
	++ --		Префіксні унарні інкремент/декремент
	& * new new[] delete delete[]		Дії з вказівниками, динамічними змінними
4	.* ->	Л-П	Розіменування члена класу
5	* / %		Арифметичні
6	+ -		
7	>> <<		
8	< > <= >=		Порівняння
9	== !=		
10	&		
11	^		Побітові
12			
13	&&		
14			Логічні
15	=	П-Л	
	+= -= *= /= %= &= ^=  = <<= >>=		Присвоєння
16	,		Об'єднання виразів у групу





# Задача

- *Задано два дійсних числа – сторони трикутника, і ціле число – кут у градусах між ними. Обчисліть периметр, площу трикутника, радіуси його вписаного та описаного кіл.*



$$S = \frac{1}{2}ab\sin\gamma \quad P = a + b + c \quad c^2 = a^2 + b^2 - 2ab\cos\gamma$$

$$R = \frac{abc}{4S} = \frac{c}{2\sin\gamma} \quad r = \frac{2S}{P} \quad \gamma_{\text{рад}} = \pi \times \frac{\gamma_{\text{град}}}{180^\circ}$$