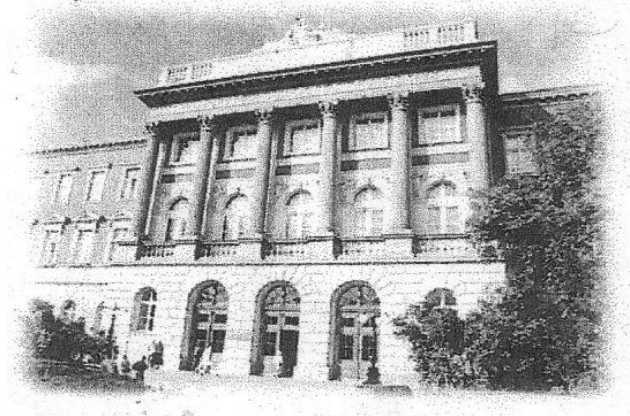


Міністерство освіти і науки України
Національний університет «Львівська політехніка»

L I T T E R I S E T A R T I B V S



Знайомство з середовищем розробки Dev C++. Створення та виконання простої програми. Програмування алгоритмів лінійної структури. Обчислення арифметичних виразів з використанням математичних функцій мови C.

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт №№ 1, 2
з дисципліни «Основи інформатики та програмування»
для студентів спеціальності 105 – "Прикладна фізика та наноматеріали"
для першого (бакалаврського) рівня освіти.

Затверджено
на засіданні кафедри
обчислювальної математики
та програмування
Протокол № 10 від 07.06.2018р.

Львів – 2018

Знайомство з середовищем розробки Dev C++. Створення та виконання простої програми. Програмування алгоритмів лінійної структури. Обчислення арифметичних виразів з використанням математичних функцій мови C. Методичні вказівки до виконання лабораторних робіт №№ 1, 2 з дисципліни «Основи інформатики та програмування» для студентів спеціальності 105 – "Прикладна фізика та наноматеріали" для першого (бакалаврського) рівня освіти. / Укл.: Гнатів Л.Б., Ментинський С.М., Пелех Я.М., Угрин С.З., 2018. - 20с.

Реєстр. № 8182 від 19.06.2018

Укладачі:

Гнатів Л.Б., к. ф-м. н., доцент,
Ментинський С.М., ст. викл,
Пелех Я.М., к. ф-м. н., доцент,
Угрин С.З., асистент.

Відповідальний за випуск: Угрин С.З.

Рецензенти:

Будз І.С., к.ф-м.н., доцент кафедри ОМП,
Філь Б.М., к.ф-м.н., доцент кафедри ОМП.

Передмова

У методичних вказівках розглянуто структуру та процес створення простої програми на C++, її компіляцію та виконання в інтегрованому середовищі розробки Dev C++ та інших поширених IDE, а також основні синтаксичні конструкції мови програмування C++ і їх використання для програмування лінійних алгоритмів.

Методичні вказівки містять коротку довідкову інформацію, варіанти завдань до виконання лабораторної роботи та приклади виконання з детальним описом.

Методичні вказівки призначені для студентів спеціальності 105 – "Прикладна фізика та наноматеріали" для першого (бакалаврського) рівня освіти і укладені відповідно до робочої навчальної програми з дисципліни «Основи інформатики та програмування».

Лабораторна робота №1

Знайомство з середовищем розробки Dev C++.

Створення та виконання простої програми

Мета роботи: ознайомлення з основами роботи в деяких IDE для C++ (MS Visual Studio, Dev C++,); ознайомлення зі структурою програми на мові C ++, елементами синтаксису мови та деякими типами даних на прикладі простої програми.

Короткі теоретичні відомості

Структура найпростішої програми на мові C/C++. Продемонструємо приклад найпростішої програми на мові C ++, яка запитує у користувача два цілочисельних значення змінних a і b, аналізує їх і виводить найбільше число.

```
#include <iostream.h>

#include <conio.h>
void main()
{
    int a,b,max;
    cout << "a=";        //запрошення ввести значення a
    cin >> a;            //ввід значення змінної a
    cout << "b=";        // запрошення ввести значення b
    cin >> b;            // ввід значення змінної b
    if(a>b) max=a;      //якщо a>b то max=a
    else max=b;         //в протилежному випадку max=b
    cout << "max="<<max; //вивід максимуму
    getch();           } //зупинка до натискання клавіші
```

Перші два рядки підключають (include - включити) до тексту програми так звані заголовкові (h від header - заголовок) файли системи. У цих файлах описані системні функції і їхні аргументи. Використовуючи ці описи, компілятор перевіряє правильність виклику системних функцій. У нашому прикладі програма використовує системні функції введення (cin >>) і виводу (cout <<), опис яких знаходяться в заголовному файлі iostream.h, а також функцію очікування натискання будь-якої клавіші (getch), опис якої знаходиться в заголовному файлі conio.h. Назви заголовків файлів найчастіше утворюються від будь-яких аббревіатур англійських слів, їх корисно навчитися розуміти, а не запам'ятовувати. У нашому прикладі: io - input / output (ввід / вивід), stream (потік), con (console - пульт оператора, тобто клавіатура і дисплей). Крім цих двох файлів ми будемо використовувати бібліотеки stdio.h для форматowanego вводу та виводу даних та math.h – бібліотеку вбудованих математичних функцій.

Третій рядок містить заголовок функції main. Функція з такою назвою повинна бути в кожній програмі на мові C/C++. Саме з неї починається виконання програми, вона - головна (саме так перекладається службове слово main). Службове слово int (від integer - цілий) повідомляє, що результатом роботи функції main повинно бути ціле число, (за яким операційна система, запустивши програму main, може "вирішити", правильно чи неправильно завершилася робота програми). За загальноприйнятою угодою нульове значення, що повертається функцією main, свідчить про нормальне завершення роботи програми. Службове слово void (дослівно - порожнеча), зазначене в круглих дужках, повідомляє, що у функції main аргументи відсутні.

Текст програми (тіло функції) розміщується в фігурних дужках (4-та 15-та стрічки). В п'ятій стрічці оголошено три змінні з іменами a, b і max, які можуть набувати цілочисельних значень. Окрім типу int в C/C++ можна використовувати і інші цілі типи даних, що відрізнятимуться діапазонами допустимих значень. Для десяткових дробів у найпростішому випадку використовуватимемо тип float (короткое дійсне, 4 байти) або double.

Шоста стрічка є першою стрічкою програми, яка виконує якусь дію - вона виводить на дисплей повідомлення, що складається з двох символів (a =). Текст повідомлення обмежують подвійними лапками. Рядок 7 організовує зупинку роботи програми до тих пір, поки користувач не набере на клавіатурі якесь число і натисне клавішу Enter. Отримане значення буде сприйняте лише, якщо воно є цілим числом, і його буде направлено в змінну a. Подібним чином в рядках 8 і 9 буде організовано ввід значення числової змінної b.

У десятій стрічці організовано порівняння поточних значень змінних a і b. Якщо значення змінної a більше, то воно присвоюється змінній max, в іншому випадку в змінну max заноситься значення b.

Рядок 12 виводить на дисплей два повідомлення - текстове (max =) і числове (значення змінної max).

Звернення до функції getch (рядок 13) призводить до затримки на екрані повідомлення програми до тих пір, поки користувач не натисне будь-яку клавішу (getch - від get character, дай символ).

Основні правила написання програми на C/C++ :

- Якщо програма звертається до якихось системних функцій, то в перших її рядках обов'язково повинна бути вказівка про підключення відповідних заголовкових файлів.

- Програма може містити більш ніж одну функцію, але серед них обов'язково має бути присутня функція з іменем main.

- Кожен рядок програми, що містить оголошення чи виконувану дію, закінчується крапкою з комою.

- Тіло функції обов'язково береться у фігурні дужки (в Паскалі аналогічні функції виконували операторні дужки - begin і end).

Використання середовищ розробки (IDE). Для написання та виконання програм на будь-якій мові програмування (в т.ч. на C та C++) необхідно, принаймні, три програмні компоненти: *текстовий редактор* (наприклад, Блокнот чи Notepad++) для написання коду програми, *транслятор* (компілятор чи інтерпретатор) мови програмування для перетворення коду у програму, готову для виконання на ПК та бібліотека стандартних підпрограм (для забезпечення елементарних операцій вводу-виводу даних та доступу до ресурсів ПК). Для зручності розробників до цього базового набору додають синтаксичні аналізатори коду, програми для зв'язування скомпільованих модулів (лінкувальники та компоувальники) та інші інструменти. Перелічені вище інструменти для розробки програмного забезпечення прийнято об'єднувати у великі програмні комплекси – інтегровані середовища розробки (IDE – Integrated Development Environment). Найпоширенішим, на сьогоднішній день, середовищем розробки програм на C++ можна вважати Microsoft Visual Studio, хоча паралельно розвиваються інші проекти – NetBeans, Eclipse і ін. Для компіляції невеликих програм, наприклад, з навчальною метою не обов'язково встановлювати спеціалізоване ПЗ, для цієї мети достатньо скористатися одним з онлайн-компіляторів в мережі Internet (див. хід виконання завдання 1).

Завдання 1. Створити проект з текстом простої програми, поданим нижче, та виконати її використовуючи

- a) Онлайн-компілятор ideone.com;
- b) Opensource IDE Dev C++;
- c) Microsoft Visual C++ з IDE MS Visual Studio.

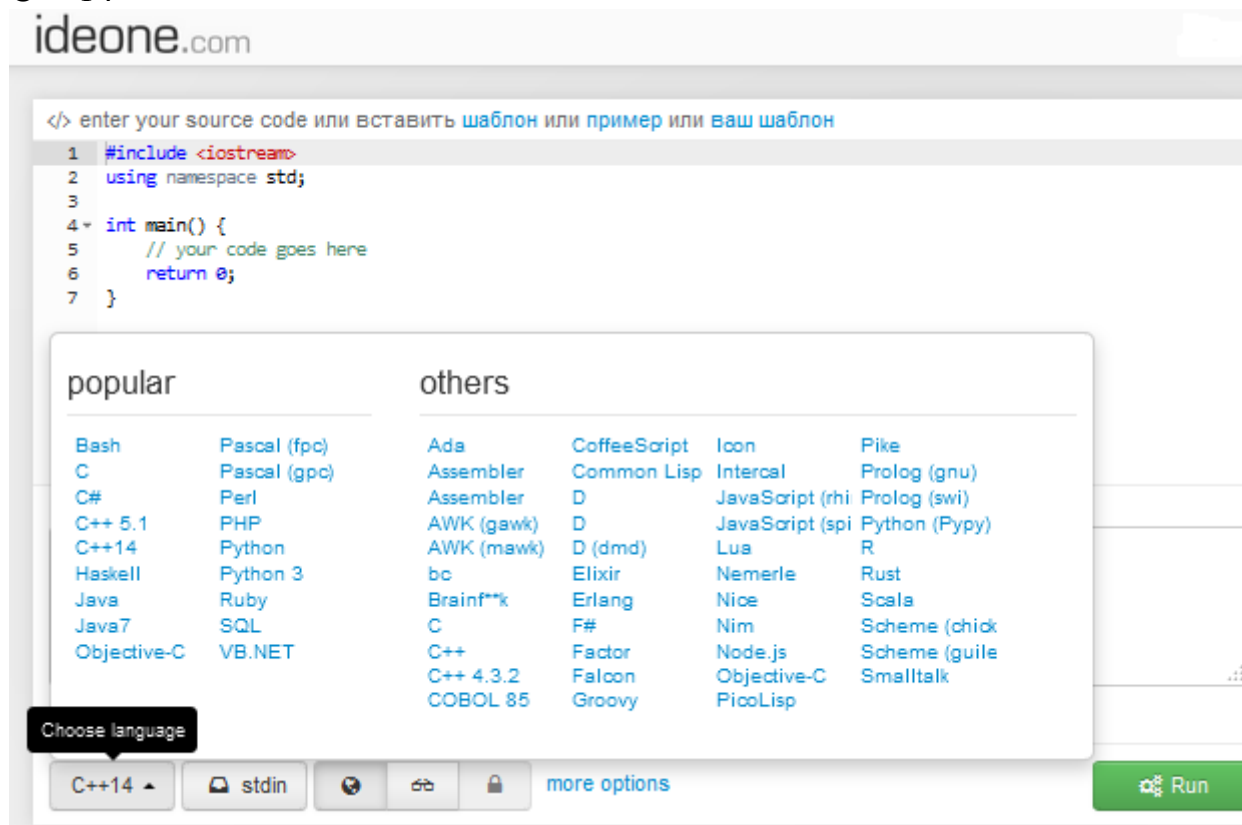
Програма з прикладу запитує два дійсних числа і виводить їх суму.
Орієнтовний код:

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      float a,b,c;
7      cin>>a>>b;
8      c=a+b;
9      cout<<"a+b="<<c;
10     return 0;
11 }
```

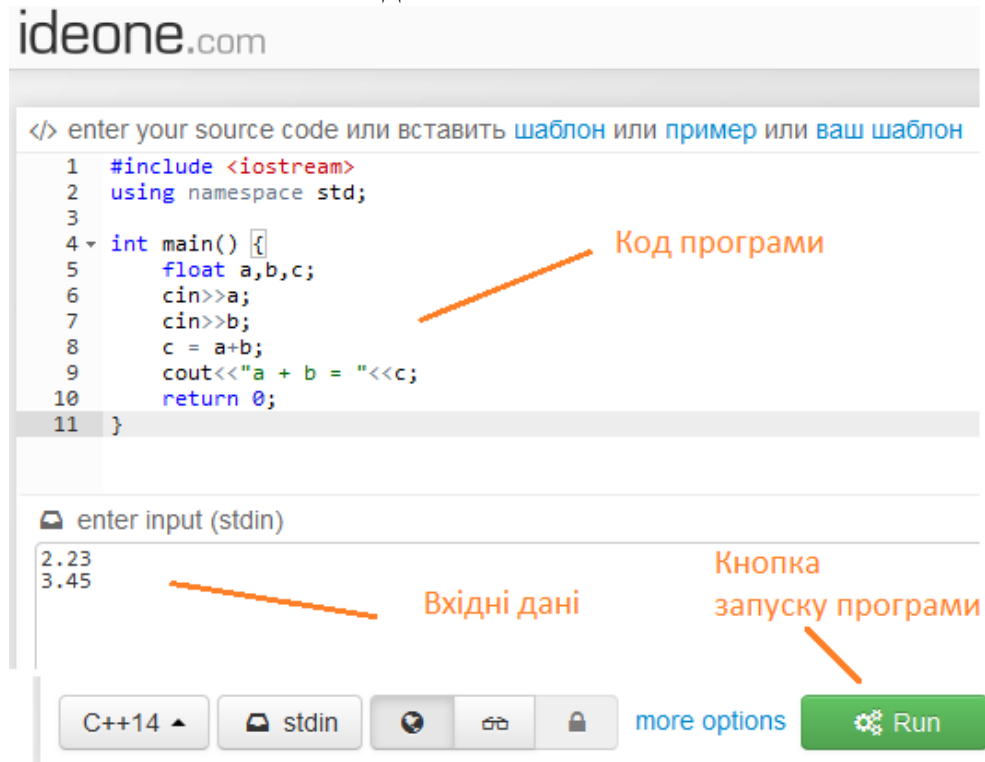
Хід виконання :

а) Онлайн-компілятор **ideone.com**.

Запускаємо браузер, переходимо за адресою **ideone.com**. Кнопкою Choose language обираємо потрібний компілятор. В нашому прикладі слід використати C++14



У полі з кодом програми замість «// your code goes here» набираємо команди нашої програми (код функції main()). В полі stdin задаємо вхідні дані – в нашому прикладі значення змінних a та b. Натискаємо кнопку Run, щоб виконати написаний код.

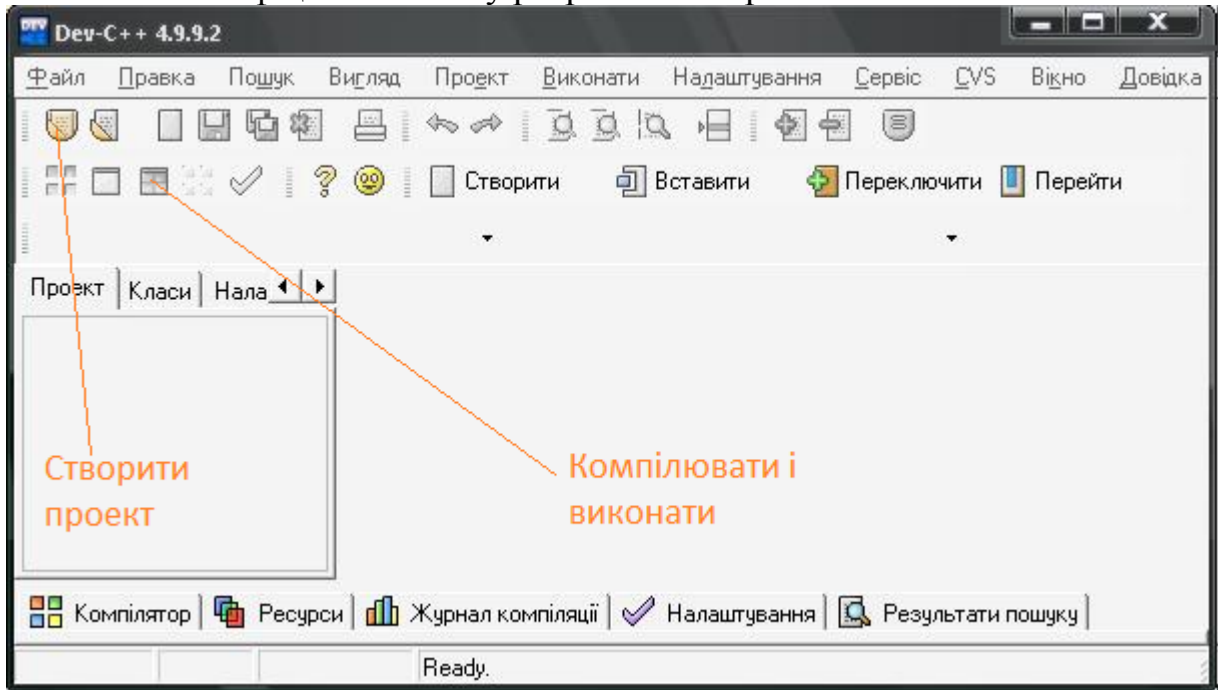


b) Opensource IDE Bloodshed Dev C++.

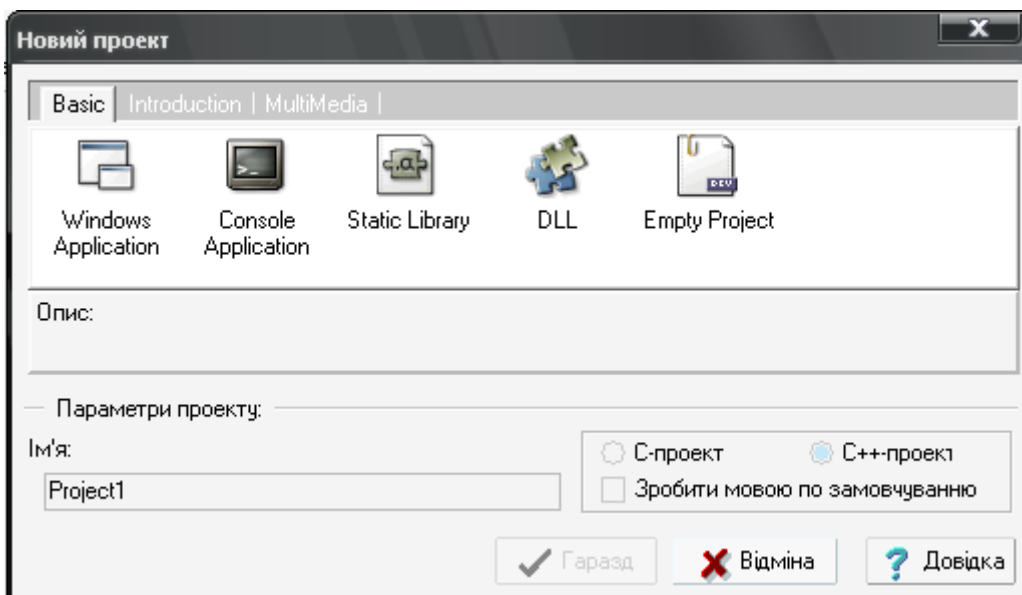
1. Запускаємо інтегроване середовище Dev C++ :

Пуск => Всі програми => Bloodshed Dev C++ => Dev C++.

Якщо програма не встановлена на комп'ютері – її дистрибутив можна завантажити з офіційного сайту розробника <http://www.bloodshed.net/>.



2. Створюємо новий проект командою Файл => Створити => Проект. У вікні «Новий проект» вибираємо Console Application (консольна програма), позначаємо мову C++ та задаємо ім'я проекту, наприклад Lab1.



3. Після підтвердження свого вибору кнопкою «Гаразд» отримаємо вікно запису проекту на жорсткий диск, в якому потрібно вибрати папку в якій буде збережено проект. Для цього можна попередньо створити папку засобами

операційної системи, або скористатися кнопкою створення нової папки у вікні збереження проекту. Щоб уникнути плутанини із файлами різних проектів, бажано для кожного проекту створювати окрему папку.

4. Після запису проекту на диск отримаємо текстовий файл із шаблоном нової програми, який редагуємо, записуючи текст нашої програми. Запустити програму на виконання можна кнопкою **Compile & Run** на панелі інструментів, або клавішею F9 на клавіатурі.

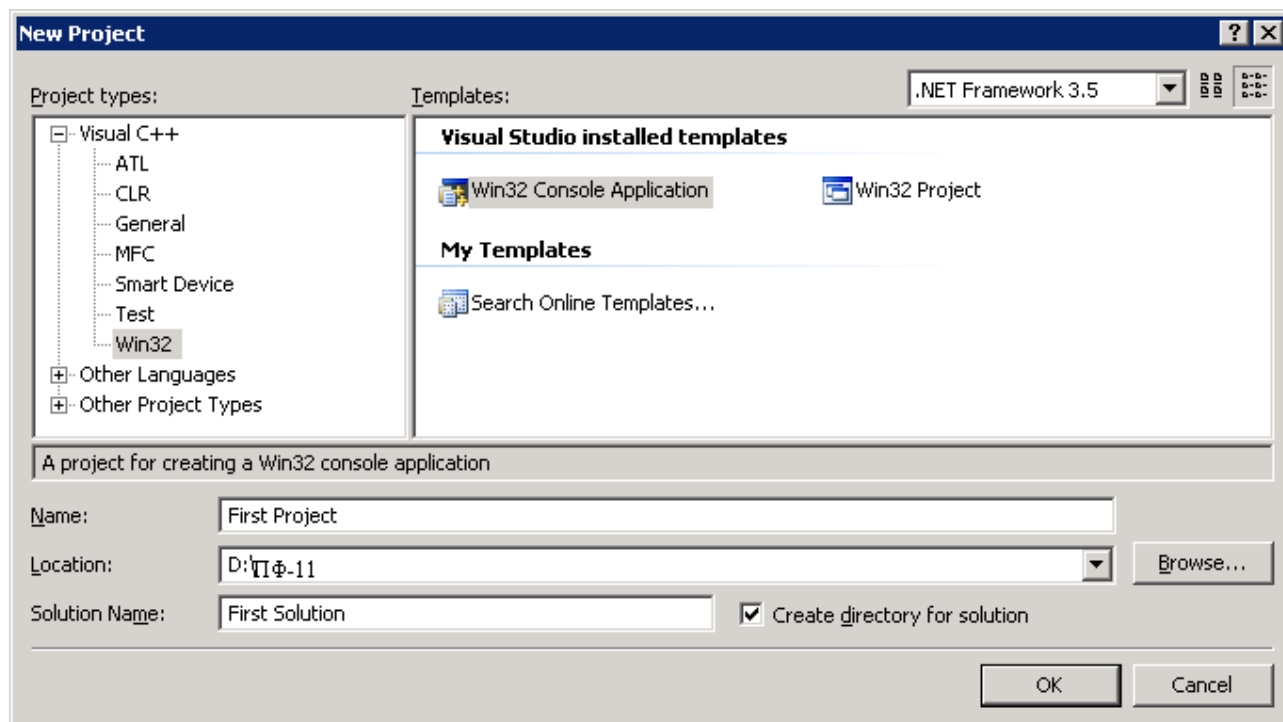
с) Середовище Microsoft Visual Studio 2008.

Запускаємо інтегроване середовище Microsoft Visual Studio:

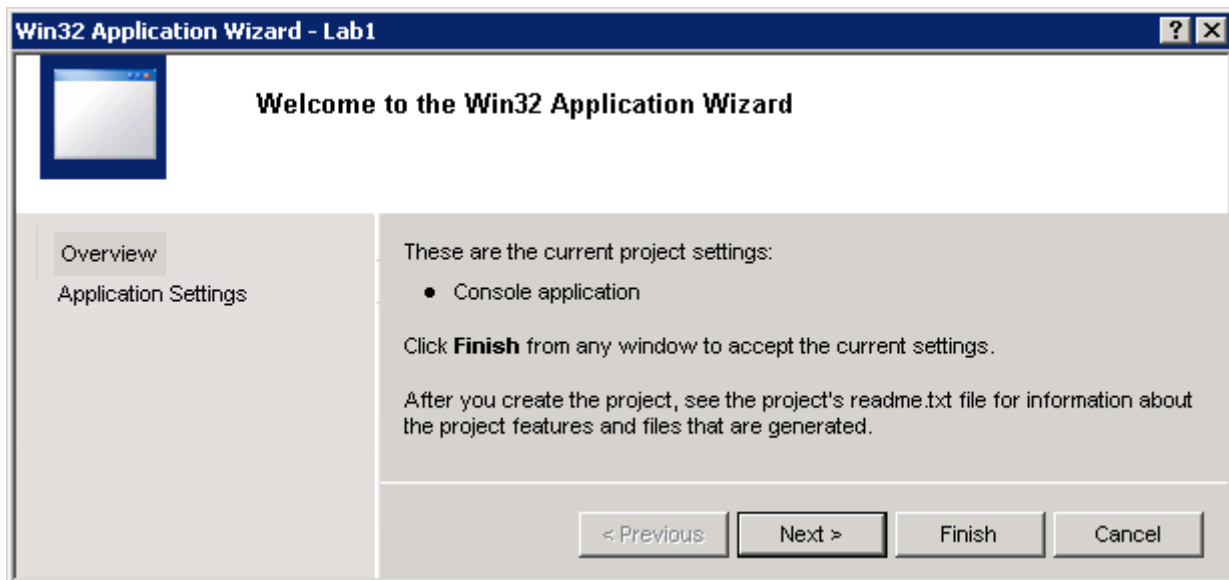
Пуск => Всі програми => Microsoft Visual Studio 2008 => Microsoft Visual Studio 2008.

Створюємо новий проект командою Файл => Створити => Проект. У вікні «Новий проект» вибираємо:

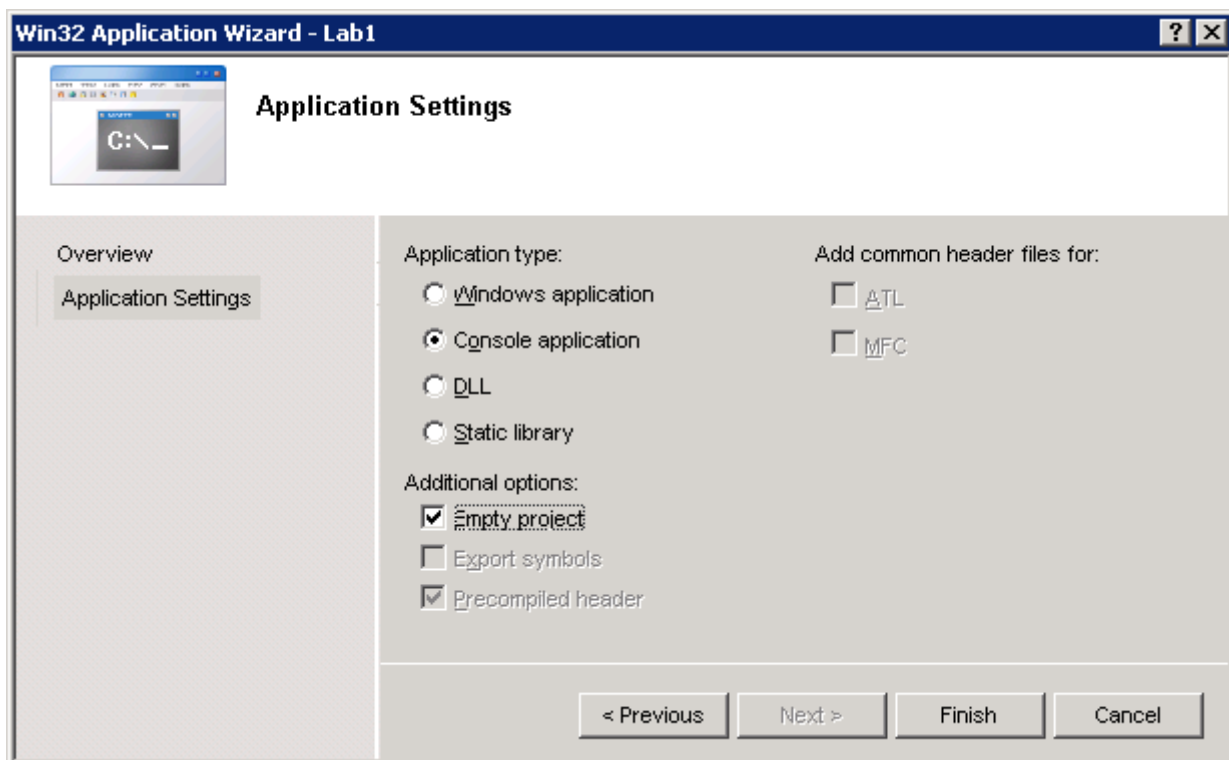
- в полі «тип проекту» – мову програмування Visual C++ та тип проекту Win 32;
- в полі «Шаблони» – шаблон Win32 Console Application;
- задаємо ім'я проекту в полі Name;
- вибираємо папку для проекту в полі Location;
- в полі Solution Name задаємо назву рішення (програми Visual Studio створюються в проекті, а проект зберігається в рішенні, одне рішення може містити багато проектів)



Задавши всі параметри, натискаємо «Ок»

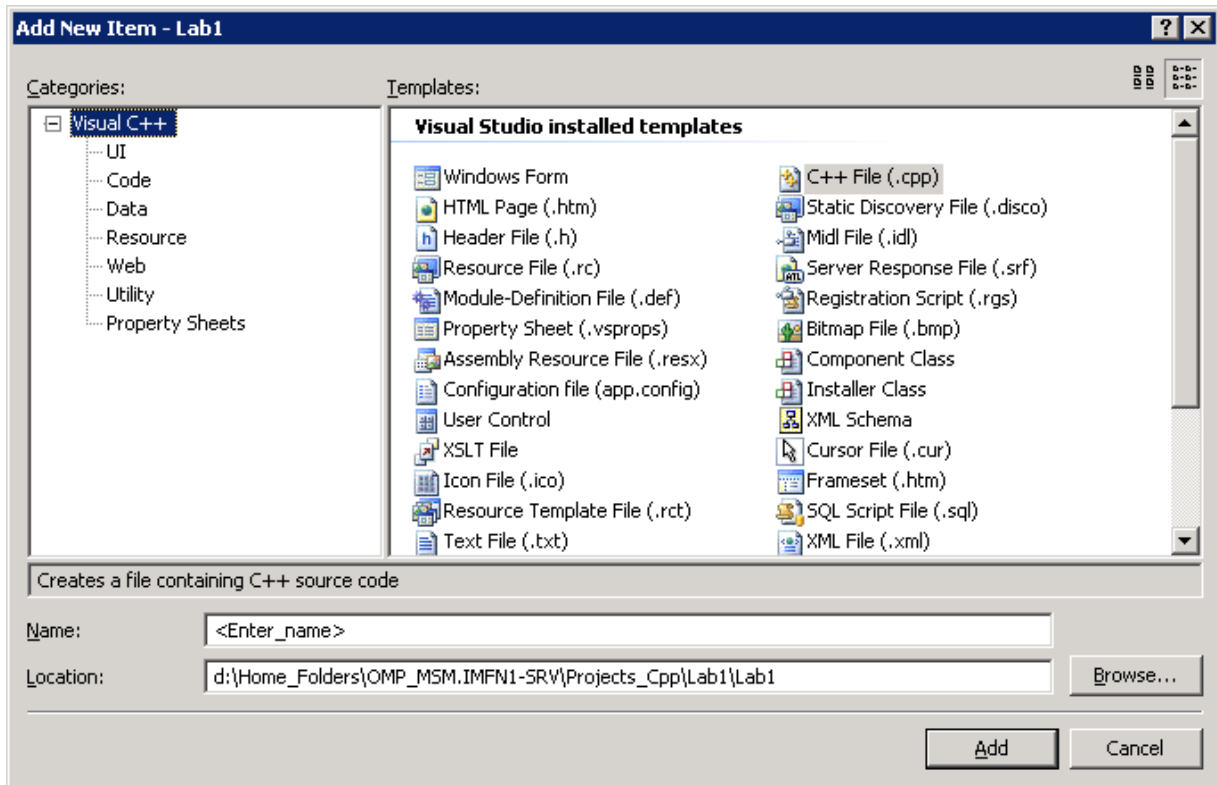


На наступному кроці майстра створення проекту слід **ОБОВ'ЯЗКОВО** перейти до наступного діалогу кнопкою «Next»!!



На цьому кроці слід вибрати пункт «**Empty project**» (порожній проект) і натискаємо «Finish».

Після створення проекту для написання коду програми до проекту слід додати новий файл командою **Add New Item** (або з контекстного меню **Add => New Item** у вікні проекту). Тип файлу повинен бути C++ **File(.cpp)**.



В полі Name вказуємо назву файлу (деякі версії дають назву Source.cpp автоматично). Далі набираємо текст нашої програми. Для запуску зручно скористатися командою **Debug=>Start Without Debugging** (комбінація клавіш Ctrl+F5).

Завдання 2.

Вважаючи усі вхідні і вихідні дані дійсними числами типу **float**, скласти програму для розв'язання задачі згідно варіанту та виконати її за допомогою будь-якого з описаних в першому завданні компіляторів. Перевірити правильність роботи програми, склавши 2-3 тестові приклади.

Варіанти завдань:

1. Дано два кола із загальним центром і радіусами R_1 і R_2 ($R_1 > R_2$). Знайти площі цих кіл S_1 та S_2 , а також площу S_3 кільця, зовнішній радіус якого дорівнює R_1 , а внутрішній радіус дорівнює R_2 . Як значення π використовувати 3.14.
2. Дано координати двох протилежних вершин прямокутника: (x_1, y_1) , (x_2, y_2) . Сторони прямокутника паралельні осям координат. Знайти периметр і площу даного прямокутника.
3. Знайти відстань між двома точками із заданими координатами (x_1, y_1) і (x_2, y_2) на площині.
4. Дано координати трьох вершин трикутника: (x_1, y_1) , (x_2, y_2) , (x_3, y_3) . Знайти його периметр, використовуючи формулу для відстані між двома точками на площині.
5. Дано сторони прямокутника a і b . Знайти його площу S і периметр P .

6. Дано довжини ребер A , B , C прямокутного паралелепіпеда. Знайти його об'єм та площу поверхні.
7. Дано число A . Обчислити A^{15} , використовуючи дві допоміжні змінні і п'ять операцій множення. Для цього послідовно знаходити A^2 , A^3 , A^5 , A^{10} , A^{15} . Вивести всі знайдені степені числа A .
8. Дано значення температури T в градусах Фаренгейта. Визначити значення цієї ж температури в градусах Цельсія. Температура по Цельсію T_C і температура за T_F Фаренгейтом T_F зв'язані наступними співвідношенням: $T_C = (T_F - 32) \cdot 5/9$.
9. Дано значення температури T в градусах Цельсія. Визначити значення цієї ж температури в градусах Фаренгейта. Температура за Цельсієм T_C і температура за Фаренгейтом T_F пов'язані наступним співвідношенням: $T_C = (T_F - 32) \cdot 5/9$.
10. Відомо, що X кг шоколадних цукерок коштує A гривень, а Y кг ірисок - B гривень. Визначити, скільки коштує 1 кг шоколадних цукерок, 1 кг ірисок, а також у скільки разів шоколадні цукерки дорожчі ірисок.
11. Дано два ненульових числа. Знайти суму, різницю, добуток і частку їх квадратів.
12. Дано два ненульових числа. Знайти суму, різницю, добуток і частку їх модулів.
13. Дано катети прямокутного трикутника a і b . Знайти його гіпотенузу c і периметр P .
14. Швидкість першого автомобіля V_1 км / год, другого - V_2 км / год, відстань між ними S км. Визначити відстань між автомобілями через T годин, якщо вони рухаються в одному напрямку.
15. Швидкість першого автомобіля V_1 км / год, другого - V_2 км / год, відстань між ними S км. Визначити відстань між ними через T годин, якщо автомобілі рухаються назустріч один одному.

Лабораторна робота №2

Програмування алгоритмів лінійної структури. Обчислення арифметичних виразів з використанням математичних функцій мови C.

Мета роботи: ознайомлення зі стандартними математичними функціями мови C++ та правилами запису арифметичних виразів на прикладі простої програми.

Короткі теоретичні відомості

Потоковий ввід та вивід даних. Для програмістів-початківців найпростіше організувати в програмі введення числової інформації з буфера потоку, пов'язаного з так званим стандартним пристроєм введення (`stdin`). Дані в цьому буфері виявляються в той момент, коли програма звертається до користувача і чекає закінчення набору затребуваних числових значень з клавіатури. Програма може запитати одне або кілька значень:

```
cin >> d;  
.....  
cin >> x1 >> x2 >> x3;
```

Перший рядок відповідає запиту на ввід значення єдиною змінною `r`. Наступний рядок програми - це запит на введення трьох числових значень, перше з яких буде присвоєно змінній `x1`, друге змінній `x2`, третє значення - `x3`. Кількість числових значень, які набираються користувачем в межах рядка, може виявитися як меншою, так і більшою, ніж потрібно програмі. У першому випадку продовження програми затримається до тих пір, поки користувач не введе достатню кількість значень. Якщо користувач набере забагато значень у рядку вводу, то зайві дані збережуться у буфері вводу і будуть передані програмі при виконанні наступного оператора `cin`. Тип значення яке вводиться і його величина мають відповідати типу змінної, в яку це значення буде записане. Відповідальність за коректний ввід даних покладається на користувача. Для забезпечення потокового введення до програми слід підключити заголовковий файл `iostream.h`, та вказати простір імен `std` (для компіляторів мови C++).

Для забезпечення виводу результатів роботи програми а також пояснень у вигляді фрагментів тексту зручно використовувати потоковий вивід на стандартний пристрій виводу (`stdout`), який здійснюється оператором виду:

```
cout << [змінна або текст в лапках]
```

наприклад,

```
cout << "f= " << f << endl;  
cout << "result:" << f*d;
```

При такому виводі можна не особливо піклуватися про формати виводу результатів, оскільки для кожного типу даних діють відповідні системні параметри форматування. Перехід на початок наступного рядка здійснюється

шляхом включення в список виводу керуючого символу '\n', або аналогічного йому символу закінчення стрічки позначеного константою endl:

```
cout << i+1 << f << f*d << endl;
```

Операції над даними цілих типів. З цілочисельними даними (константами і змінними) в мовах C, C++ можна виконувати звичайні арифметичні операції - додавання ($x+y$), віднімання ($z-5$), множення ($x1*x3$) і ділення (y/w). На відміну від мови Pascal тут ділення цілочисельних операндів дає цілочисельний результат. Наприклад, $5/2=2$. Для обчислення остачі від ділення цілих чисел в C/C++ використовується операція %, наприклад, $5\%2=1$.

Над цілочисельними даними можна також виконувати операції порозрядного зсуву (знаки «<<» та «>>»):

```
y=x<<3; // зсув вліво на три двійкові розряди  
z=y>>5; // зсув вправо на п'ять двійкових розрядів
```

Над однойменними двійковими розрядами цілочисельних операндів можна також виконувати логічні операції - логічне додавання «|», логічне множення «&», виключне або «^» та інвертування «~». В літературі за цими операціями закріпилася назва «побітові».

Певну допомогу при обробці цілочисельних даних можуть надати системні функції математичної бібліотеки. Прототипи цих функцій описані в заголовних файлах math.h і stdlib.h. Аргументом функцій **atoi (s)** і **atol (s)** є рядок, що представляє запис цілого числа. Кожна з цих функцій перетворить символний запис числа у відповідний машинний формат (результат atoi має тип int, результат atol - тип long) і повертає отриманий результат. Досить корисне перетворення виконують функції **itoa** і **ltoa**. Перший їхній аргумент - числове значення типу int або long. Другим аргументом є рядковий масив (або вказівник на рядок), куди записується результат перетворення. А третій аргумент, значення якого знаходиться в діапазоні від 2 до 36, визначає основу системи числення, в яку перетвориться значення першого аргументу.

У ряді математичних алгоритмів, що використовують імовірнісні методи, а частіше - в ігрових програмах активно використовуються різні генератори випадкових чисел. Функція **random (N)** при кожному повторному зверненні до неї видає чергове випадкове число з діапазону від 0 до N-1. Ці числа мають рівномірний розподіл ймовірностей.

Операції над числовими даними дійсних типів. Для обчислень з дійсними даними дозволяється використовувати лише чотири арифметичні операції - додавання (+), віднімання (-), множення (*) і ділення (/). В арифметичному виразі можуть зустрічатися операнди різних типів, тому важливо знати, як компілятор визначає тип результуючого значення. Правило це досить просте,

але воно містить підводні камені. Арифметичні дії виконуються відповідно загальноприйнятих правил:

- спочатку виконуються дії в самих внутрішніх дужках;
- всередині дужок порядок дій визначається пріоритетом операцій - спочатку одномісні операції типу зміни знака, інвертування і обчислення функцій, потім множення і ділення і, в останню чергу, додавання і віднімання;
- тип результату функції описаний або в системних заголовних файлах, або в програмі користувача;
- тип операнда, добутку співмножників, визначається типом «найширшого» співмножника (тобто допускає найширший діапазон представлення даних);
- тип виразу, що складається з доданків, визначається типом «найширшого» доданка.

Незважаючи на простоту і природність описаних вище правил, результати обчислення деяких виразів можуть завдати клопотів не дуже уважним програмістам. Наприклад, не викликає сумніву, що тип виразу $5/2 + 1.0$ повинен бути дійсним. Компілятор орієнтується на тип числової константи 1.0, яка за правилами системи програмування перетвориться в машинний формат довгого дійсного числа (тобто має тип double). А ось результат обчислення даного виразу дорівнює 3.000000, бо тип кожного доданка визначається незалежно від типів інших доданків. Перший доданок представлено часткою двох цілих констант, тому його тип теж цілий, тобто результат ділення дорівнює 2, а не 2,5. Потім значення всіх доданків приводяться до типу double, і підсумковий результат дорівнює 3,0.

Список деяких математичних функцій наведено в таблиці (хідер math.h.)

Функція	Результат	Функції	Результат
acos(x)	arccos x	log(x)	ln x
asin(x)	arcsin x	log10(x)	lg x
atan(x)	arctg x	pow(x, y)	x^y
atan2(x, y)	arctg (y/x)	pow10(int p)	10^p
ceil(x)	заокруглення "зверху"	floor(x)	Заокруглення "знизу"
cos(x)	cos x	sinh(x)	sh x
cosh(x)	ch x	sqrt(x)	квадратний корінь з x
exp(x)	e^x	tan(x)	tg x
fabs(x)	x	tanh(x)	th x
		sin(x)	sin x

У заголовному файлі math.h наводяться визначення іменованих констант, якими корисно скористатися в своїх програмах

Константа	Значення	Константа	Значення
M_PI	π	M_E	$e=2.718\dots$
M_PI_2	$\pi/2$	M_LOG2E	$\log_2 e$
M_PI_4	$\pi/4$	M_LOG10E	$\lg e$
M_2_PI	$2/\pi$	M_LN2	$\ln 2$
M_1_SQRTPI	$1/\sqrt{\pi}$	M_LN10	$\ln 10$
M_2_SQRTPI	$2/\sqrt{\pi}$	M_SQRT2	$\sqrt{2}$

Для використання математичних констант перед директивою підключення хідер-файла math.h слід задекларувати їх використання директивою

```
#define USES_MATH_DEFINES
```

• Завдання

Проаналізувати варіант завдання. Якщо арифметичний вираз громіздкий, розбити його на частини. Скласти блок-схему та програму для обчислення значень функцій відповідного варіанту. Реалізувати програму на ПК. На друк вивести значення вихідних даних і результати обчислень, супроводжуючи вивід найменуванням змінних. Змінну U вивести з трьома знаками після коми. Для змінної V вивести окремо цілу і дробову частини.

Варіанти завдань

№	РОЗРАХУНКОВІ ФОРМУЛИ	Значення вихідних даних
1	$U = 4 \sqrt[4]{\frac{\sqrt[5]{B} * \sin 1}{A * B * \ln 3}}$ $V = B * \lg(3 * 2 * \sin 3 - 3 \sin 2)$ $Z = e^{2 \arctg X + 3 \arcsin^3 Y} + X^A$	A=1.3; B=4.627 X=2.4 Y=0.7
2	$U = \frac{\sqrt[3]{A^2 * \ln 2}}{B * \sqrt[5]{C} * \cos 1}$ $V = A * \arcsin\left(-\frac{1}{4} * (\arctg 2 + \arctg 3)\right)$ $Z = \lg 3 * \lg(X + Y) + A^Y \operatorname{tg} X$	A=123.4 B=1.124 C=1.24 X=7.4 Y=0.12
3	$U = A^A \sin 3 - B^B \cos 3$ $V = C * \operatorname{tg} 4 * C^{2^{\frac{1}{1 - \lg^4 \sqrt{4}}}}$ $Z = \sqrt{ X^3 e + Y e^{-1} }$	A=0.461 B=0.356 C=99 X=2.4 Y=4.9
4	$U = A^B \sin 2 + (2A)^{\frac{B}{2}} \sin 3 - (3e)^{-1} \sin 2.3$	A=2.564 B=0.754

	$V = Y * \frac{\pi}{3} \lg 2 + \sin(\arccos(-\frac{1}{7}) - \arccos(-\frac{13}{14}))$ $Z = \frac{X - Y}{\operatorname{tg}^3 \frac{X}{2}} * \frac{1}{\sqrt{3X^2 + 4Y^2}}$	$X=1.7$ $Y=5.4$
5	$U = A^B \log_2 5 + C * \log_3 7$ $V = C * \operatorname{tg} 6 * e^{-(\arccos \sqrt{\frac{2}{3}} - \arccos \frac{\sqrt{6}+1}{2\sqrt{3}})}$ $Z = \lg^4 X - Y $	$A=0.171$ $B=1.163$ $C=2.526$ $X=7.6$ $Y=0.3$
6	$U = \sqrt{\frac{12.4e + 0.6e^{-1} * \sqrt[3]{A}}{B(\ln 3 + \sin 1)}}$ $V = Y * \operatorname{tg}(5 \operatorname{arctg} \frac{\sqrt{3}}{7} - \frac{1}{4} \arcsin \frac{\sqrt{3}}{5})$ $Z = \frac{3X + 2Y}{(7X^2 + Y^2)Y^X} * \ln^3 X + Y $	$A=0.389$ $X=0.7$ $Y=2.8$
7	$U = \frac{A^2}{\sqrt[3]{B}} * \sin 3 + \frac{\sqrt[4]{B}}{C^2 A^B} \operatorname{tg}^3 3$ $V = e^2 * \sin(3 \operatorname{arctg} 2\sqrt{3} + 2 \arccos \frac{1}{7})$ $Z = \ln(X + 5 Y)$	$A=1.592$ $B=0.382$ $C=0.5348$ $X=1.2$ $Y=21.5$
8	$U = \frac{\sqrt[3]{A \ln 3} - \sqrt{B \ln 5}}{3A^{1.1}}$ $V = B * \operatorname{tg} 6 * \cos(2 \arcsin \frac{\sqrt{3}}{4} + \arccos(-\frac{7}{9}))$ $Z = \sqrt{XY^2 - Ye^{-2}}$	$A=79.836$ $B=156.374$ $X=2.7$ $Y=3.1$
9	$U = \sqrt{\frac{Ae + 0.6e^{-1} \sqrt[3]{B}}{C * (\ln 3 + \sin 1)}}$ $V = A * \operatorname{tg} \left 5 \operatorname{arctg} \frac{\sqrt{3}}{5} - \frac{1}{7} \arcsin \frac{\sqrt{3}}{5} \right $ $Z = \frac{3X + 2Y}{(X^2 + Y^2)BC}$	$A=12.4$ $B=0.548$ $C=0.389$ $X=1.2$ $Y=2.4$
10	$U = \frac{\sqrt{ B - A } \lg X}{C(\cos X + \cos Y)}$ $V = A * \operatorname{tg} \left 7 \operatorname{arctg} \frac{\sqrt{6}}{5} - \frac{1}{9} \arccos \frac{\sqrt{2}}{5} \right $ $Z = \frac{5X - 3Y}{(X^3 + Y^3)B^C} * X^4$	$A=10.7$ $B=5.4$ $C=0.89$ $X=1.5$ $Y=2.7$
11	$U = (A \ln 3)^{B \ln 5}$ $V = X * \operatorname{arctg}((\sin 1 * \ln \sqrt{3 \ln 3}))$	$A=0.273$ $B=1.573$ $X=2.4$

	$Z = \frac{7X - 5Y}{2X^2 + 3Y^2} \arcsin^3 A + \lg \operatorname{tg} B $	Y=1.7
12	$U = \frac{\sqrt[3]{A^2 \ln 2}}{B\sqrt[5]{C} \cos 1}$ $V = A * \arcsin(-\frac{1}{4}(\operatorname{arctg} 2 + \operatorname{arctg} 3))$ $Z = e^{- X+2Y } + \operatorname{tg}^4 A^C$	A=123.4 B=1.124 C=0.24 X=1.3 Y=0.5
13	$U = \sqrt[3]{A} \cos 7 + \sqrt[4]{B} \sin 7$ $V = A * \cos(-\frac{1}{\sqrt{2}}(\operatorname{arctg} 3 - \operatorname{arctg} 2))$ $Z = \lg 3 * \ln(X + Y) + \arccos^2(\frac{2}{3 \operatorname{tg} 3 * A} B^A)$	A=2.324 B=0.836 X=2.7 Y=0.3
14	$U = A^B * \cos 5 - 2^{-C} * \sin 5$ $V = A * \sin(3 \operatorname{tg} 4 * \arccos \frac{\sqrt{3}}{4})$ $Z = \frac{e^{-\frac{X-Y}{X^2+Y^2}} * \sqrt[3]{AB}}{3 \lg^2 AX}$	A=81 B=0.48 C=-0.37 X=2.5 Y=0.63
15	$U = \sqrt{\frac{A\sqrt[3]{B}}{3C^4(e^2 + e^{-2})}}$, $V = A * \arcsin^2 \operatorname{tg} 4XY$ $Z = \lg \left A^{-\frac{3}{4}} - 2\sqrt{2} \right + \operatorname{arctg} \frac{X}{3} + \frac{1}{3} \arccos X$	X=0.9 Y=0.17 A=2.591 B=0.0839 C=1.114

Приклад Варіант 15

```
#include <iostream>
#include <math.h>
using namespace std;

int main(void)
{
    float a, b, c, x, y;
    float u, v, z1, z2, z;
    int v1;
    cout << "Vvedit znachennja a, b, c ";
    cin >> a >> b >> c;
    cout << "Vvedit x, y ";
    cin >> x >> y;
    u = sqrt((a * pow(b, 1/3))/(3 * c * c * c * c * 2 * cosh(2)));
    v = a * pow(asin(tan(4 * x * y)), 2);
    z1 = log10(fabs(pow(a, -3/4) - 2*sqrt(2)));
    z2 = atan2(3, x) + 1/3 * acos(x);
    z = z1 + z2;
    v1 = ceil(v);
```

```

    system("cls");
    cout << "Input data: \n";
    cout << "a = " << a << "\nb = " << b << "\nc = " << c;
    cout << "\nx = " << x << "\ny = " << y;
    cout << "\nResult: \n";
    cout << "u = ";
    cout.width(10);
    cout.precision(3);
    cout << u;
    cout << "\nv = " << v << "\nz = " << z;
    cout << "\nv_int = " << v1 << "\nv_frac = " << v - v1;
    system("pause");
    return 0;
}

```

Рекомендована література:

1. Програмування на C(C++). Парадигма процедурного програмування. Навчальний посібник. /Гнатів Б.В., Гнатів Л.Б. . – Львів: Видавництво «Растр-7», 2017. – 264 с.
2. Основи програмування на C/C++. Конспект лекцій з курсу «Основи інформатики і програмування, частина 2» спеціальності 105 – «Прикладна фізика та наноматеріали» для першого (бакалаврського) рівня освіти/ Укл.: Ментинський С.М., 2016. – 140 с
3. Програмування мовою C: навчальний посібник для вузів / Зореслава Ярославівна Шпак . — Львів: Оріяна-Нова, 2006 . — 431 с.
4. Крєневич, А.П. C у задачах і прикладах : навчальний посібник із дисципліни "Інформатика та програмування" / А.П. Крєневич, О.В. Обвінцев. – К. : Видавничо-поліграфічний центр "Київський університет", 2011. – 208 с.
5. Стивен Пратта Язык программирования C. Лекции и упражнения - К.: Диасофт, 2002.

НАВЧАЛЬНЕ ВИДАННЯ

Знайомство з середовищем розробки Dev C++. Створення та виконання простої програми. Програмування алгоритмів лінійної структури. Обчислення арифметичних виразів з використанням математичних функцій мови C.

МЕТОДИЧНІ ВКАЗІВКИ до виконання лабораторних робіт №№ 1,2 з дисципліни «Основи інформатики та програмування»

для студентів спеціальності 105 – "Прикладна фізика та наноматеріали" для першого (бакалаврського) рівня освіти.

Укладачі:

Гнатів Л.Б., к. ф-м. н., доцент,
Ментинський С.М., ст. викл,
Пелех Я.М., к. ф-м. н., доцент,
Угрин С.З., асистент.

Комп'ютерне складання: Ментинський С.М., ст. викл.