**Практична робота №2. Елементи мови С++. Знайомство з середовищем програмування**

**Мета:** навчитися складати та виконувати просту програму на С++, використовуючи обраний компілятор (C++SHELL, GBD – URL: https://www.onlinegdb.com/).

**Завдання:**

1. Ознайомитися з теоретичними відомостями.
2. Запустити в Інтернет-браузері он-лайн компілятор C++Shell за посиланням <http://cpp.sh/> або використати інший доступний вам компілятор С++, наприклад GBD – URL: https://www.onlinegdb.com/

В браузері набрати // <http://cpp.sh/> для виклику он-лайн компілятору C++SHELL, буде надане вікно (рис.1)

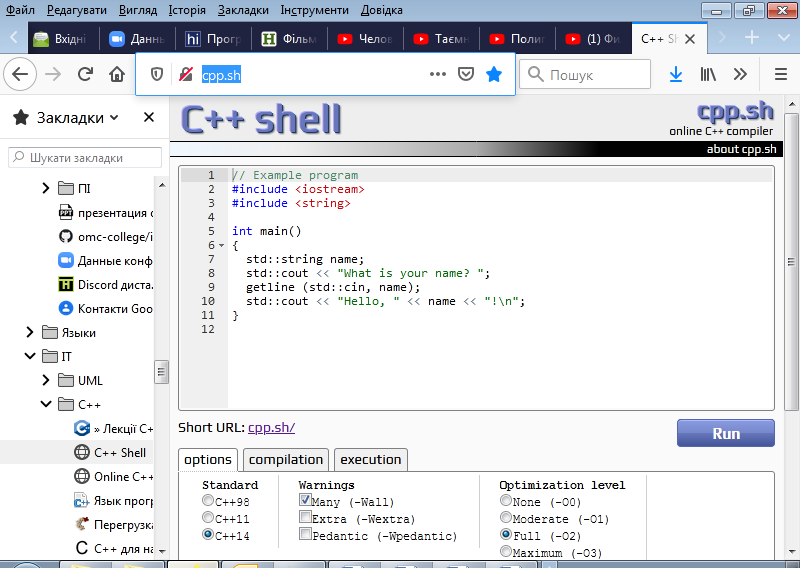


Рисунок 1 – вікно он-лайн компілятору C++SHELL

1. В цьому вікні відкоригувати текст відповідно до наданого на рис.2, вказавши в першому рядку своє прізвище.

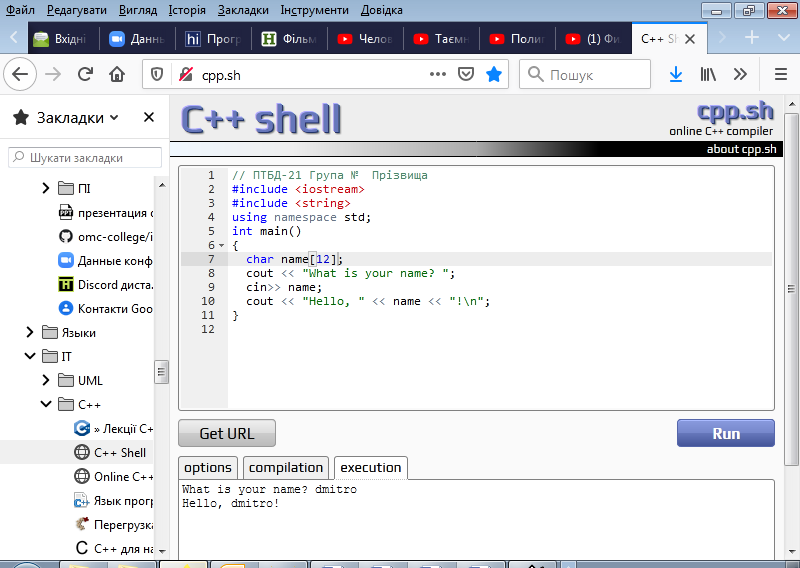


Рисунок 2 – текст зміненої програми

1. Натиснути клавішу **Run** для запуску на виконання і у закладці **execution** ввести ім’я за прикладом на рис.3.

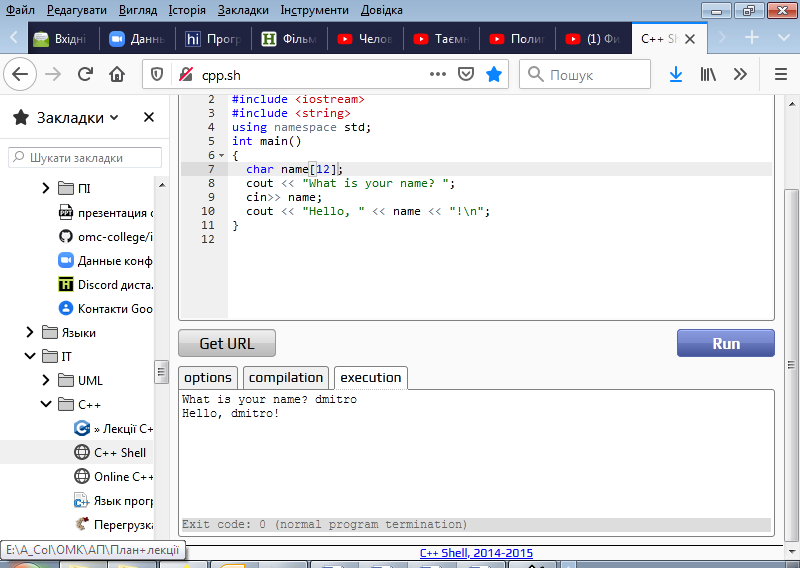


Рисунок 3 – виконання програми

1. До тексту програми додайте цілу змінну age (вік), а також після останнього оператору виведення (cout << "Hello, " << name << "!\n";) додайте оператор виведення тексту "How old are you? "
2. Після цього введіть значення змінної age оператором

cin>> age;

і виведіть вік оператором cout << "You are "<< age << " years old";

1. Зробіть скріншот програми і результатів виконання для збереження в звіт до цієї практичної роботи.
2. Замініть текст програми, починаючи з 5-го оператору: введіть другу програму з Лекції №3 з урахування змін на рис. 4.

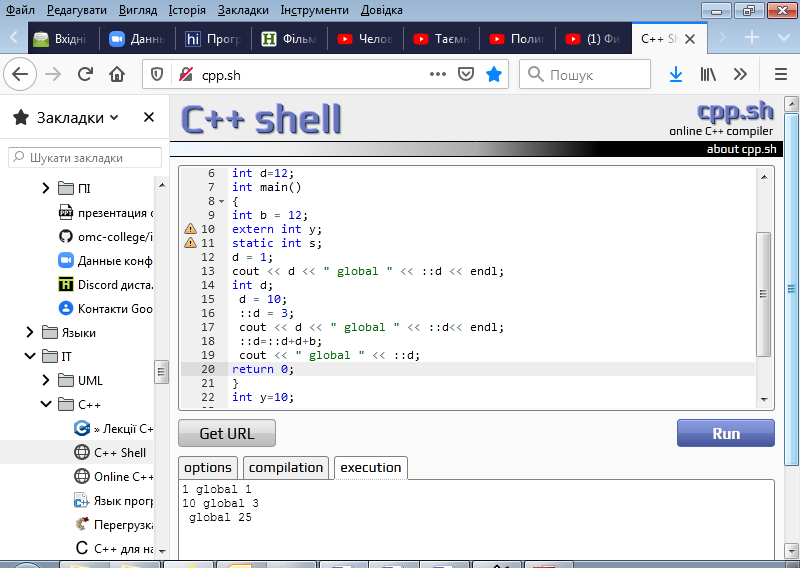
****

Рисунок 4 – друга програма з Лекції №3 зі змінами

1. Зробіть скрінщот програми і результатів виконання.
2. Оформіть практичну роботу, надавши в ній скріншоти виконаного завдання, а також надай те письмові відповіді на запитання:
3. Що на Вашу думку означають позначки у вигляді жовтого трикутника зі знаком оклику всередині і чому вони надаються?
4. Чому в першому випадку виводиться двічі значення "1"?
5. Для чого на Вашу думку записана лексема endl?
6. Звіт надсилати на електронну адресу викладача [**t.i.lumpova@gmail.com**](mailto:t.i.lumpova@gmail.com)у вигляді файлу з іменем у форматі

**<Номер групи><Номер лабораторної><Прізвище англійською>**

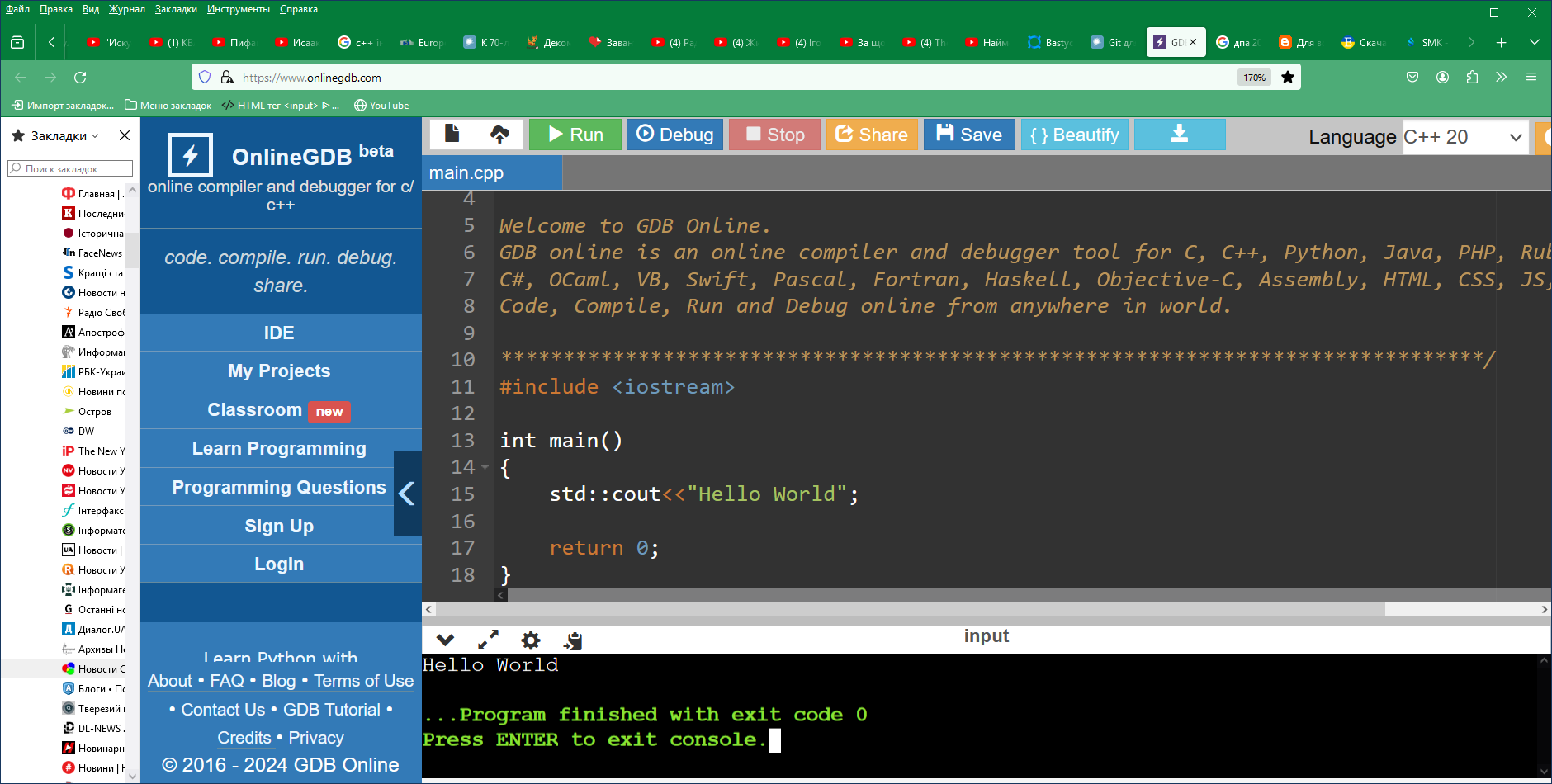
Наприклад, OPI41-01Ivanov.cpp.

**Строк відсилки ЛР для МІВТ/ ЕТ-41 24.02.2024**

Всі запитання, що виникнуть, надсилайте на електронну адресу викладача.

В темі листа вказати, номер групи, прізвище студента та номер ПР як "ПР№2" В темі листа вказати, номер групи, прізвище студента, номер ПР та фразу "Запитання".

Можна також використати безкоштовний он-лайн компілятор **GBD** за посиланням https://www.onlinegdb.com/



Виклик

виконання

**Теоретичні відомості**

**Директиви підключення ресурсів**.

Служать для підключення до тексту програми текстів *заголовних файлів* (мають стандартне розширення .h). ***Заголовний файл*** – це файл, що містить описи деяких функцій та оголошення типів даних і констант, які використовуються цими функціями. Список заголовних файлів визначається стандартом мови.

Існують два способи підключення директиви #include:

1. # include <ім'я> - підключення стандартних заголовних файлів;
2. # include “ім'я” - підключення заголовних файлів користувача.

Різниця між ними полягає в тому, де препроцесор буде шукати файли-ресурси, які необхідно підключити для повної підготовки коду програми до компіляції: якщо ім’я файлу взяте в кутові дужки <…>, то пошук буде вестись в наперед визначених каталогах; якщо ім’я взяте в лапки, то препроцесор шукає його в поточному каталозі, де міститься і основний файл компілювання.

Слід зазначити, що багато стандартних заголовних файлів має порожнє розширення, для решти - традиційно використовують розширення h. Наприклад,

#include <iostream> *// підключення засобів потокового введення-виведення*

#include < stdio.h> *// підключення засобів форматованого введення-виведення*

Основні заголовні файли, які входять до складу стандартної бібліотеки, подані в табл.1.

Таблиця 1. Основні заголовні файли стандартної бібліотеки С/С++

| *Заголовний файл* | *Вміст* |
| --- | --- |
| <iosream.h> | прототипи для функцій стандартних вхідних і вихідних потоків |
| <iomanip.h> | прототипи функцій для маніпуляторів, що дозволяють форматувати потоки даних |
| <cmath.h> | прототипи функцій математичної бібліотеки |
| <stddef.h> | загальні визначення типів, що використовуються в С/С++ для виконання деяких обчислень |
| <stdio.h> | прототипи для функцій вводу/виводу стандартної бібліотеки і інформація, що ними використовується |
| <stdlib.h> | прототипи функцій для перетворення чисел в текст і тексту в числа, прототипи функцій розміщення в пам’яті, генерації випадкових чисел і інших сервісних функцій |
| <time.h> | прототипи функцій і типи для функцій управління часом і датою |

Розглянемо найпростіший приклад програми на C/*С++* :

**#include <iostream> *// директива препроцесора***

**using namespace std; *// простір імен***

**int main() *// головна функція***

**{ cout << "Hello,world\n"; *// виведення повідомлення на екран***

**return 0; *// повернення результату (вихід із функції)***

**}**

Рядок #include (включити) означає, що препроцесор перед компіляцією програми має включити в неї вміст заголовного файлу iostream.h. У цьому файлі оголошено засоби потокового введення та виведення. Без включення цього файлу ім’я cout буде невизначеним і компілятор повідомить про цю помилку.

У другому рядку розташовано інструкцію компілятору «використати простір імен std». Не пояснюючи значення слів «простір імен», скажемо лише, що простір імен std є стандартним. У сучасних системах програмування мовою С++ у ньому описано всі бібліотечні засоби «останнього покоління». Проте не всі компілятори розуміють цю інструкцію, тому її можна писати не завжди.

Частина програми

**int main()**

**{ ...**

**}**

визначає головну функцію main.

Операція << ("помістити в") пише свій перший аргумент у другий (рядок "Hello,world\n" у стандартний потік виведення cout). У даному рядку символ “\n“ є управляючим символом переходу на новий рядок.

За інструкцією system("pause"); виконання програми призупиняється і на екрані з’являється повідомлення, що треба натиснути будь-яку клавішу. Після її натискання програма завершується. Завдяки використанню управляючого символу “\n“, повідомлення виводиться в новому рядку. Якби його не було, повідомлення з’являлося б відразу після слів Hello, world!.