Міністерство освіти та науки України

Інститут «Телекомунікації, радіоелектроніки та радіотехніки»

Звіт до лабораторної роботи № 7

З дисципліни «Програмування частина 2»

Арифметичні операції та вирази мови C

Виконав студент групи АП-12:

Дроф’як Святослав

Перевірив викладач:

Пелех Н.В.

Львів 2024

Мета роботи: ознайомитися з синтаксисом арифметичних операцій, їх пріоритетом застосувань, навчитися їх використовувати для обчислень математичних виразів.

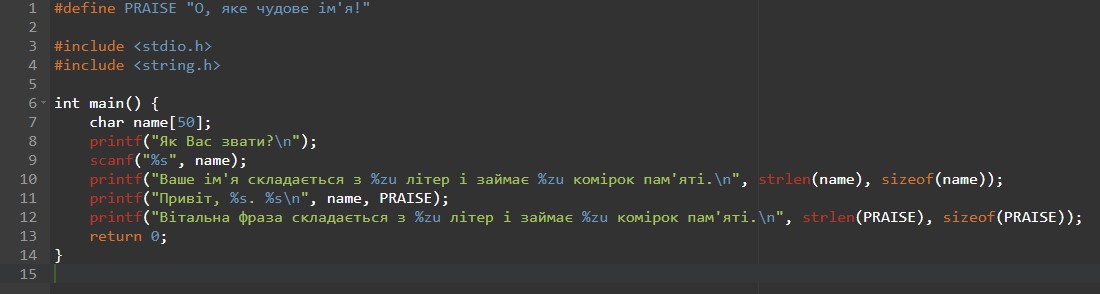
# Хід роботи

1. Ознайомитися з теоретичними відомостями.
2. Здійснити виконання усіх прикладів, представлених у теоретичних відомостях, після чого представити скріни їх коду та результати їх виконання у звіті.
3. Набрати текст нижченаведеної програми, виправити усі синтаксичні помилки, здійснити її компіляцію.
4. Скрін коду набраної програми та її результати роботи (після усіх виправлень) представити у звіті.
5. Виконати нижченаведені програми. Скріни коду набраних програм та її їх результати роботи представити у звіті.
6. Оформити звіт.

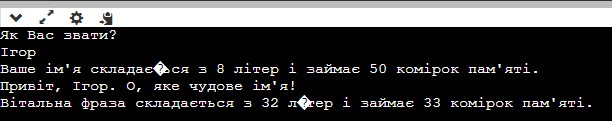
**Завдання 1**

**Реалізація:**

Програмна реалізація мовою С



Код завдання 1, приклад 1



Скріншот виконання коду, приклад 1

Ця програма на мові програмування C запитує ім’я користувача, а потім виводить кількість літер у введеному імені, розмір виділеної пам’яті для змінної name, вітає користувача, використовуючи введене ім’я та фразу, визначену в PRAISE, і виводить кількість літер та розмір пам’яті, який займає ця фраза. Ось короткий опис кожного кроку:

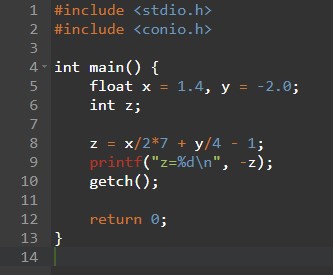
1. #define PRAISE "О, яке чудове ім'я!" - визначає константу PRAISE.
2. char name[50]; - оголошує масив символів name розміром 50 для зберігання імені.
3. scanf("%s", name); - читає введене користувачем ім’я.
4. strlen(name) - визначає довжину введеного імені.
5. sizeof(name) - визначає розмір масиву name, який завжди буде 50, оскільки це розмір масиву.
6. printf("Привіт, %s. %s\n", name, PRAISE); - виводить вітальне повідомлення з іменем та фразою PRAISE.
7. strlen(PRAISE) і sizeof(PRAISE) - визначають довжину та розмір пам’яті фрази PRAISE. Зверніть увагу, що sizeof(PRAISE) включає нульовий символ на кінці рядка, тому буде на одиницю більше, ніж strlen(PRAISE).

Програма призначена для демонстрації роботи з рядками та масивами у мові C.

**Завдання 2**

**Реалізація:**

Програмна реалізація мовою С



Код завдання 2, приклад 1



Скріншот виконання коду, приклад 1

Ця програма на мові C виконує наступні дії:

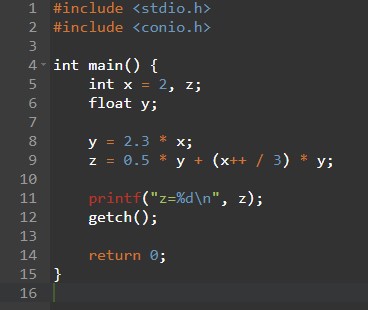
1. Оголошує змінні x та y типу float і присвоює їм значення 1.4 та -2.0 відповідно.
2. Оголошує змінну z типу int.
3. Виконує обчислення за формулою z = x/2\*7 + y/4 - 1 і присвоює результат змінній z.
4. Виводить на екран значення -z (знак мінус перед z означає, що буде виведено протилежне значення z).
5. Функція getch() чекає на натискання клавіші користувачем перед закриттям програми.

Однак, важливо зазначити, що вираз x/2\*7 + y/4 - 1 спочатку обчислюється як float, а потім конвертується до int перед присвоєнням z, що може призвести до втрати дробової частини. Також, через використання int для z, результат буде округлений до цілого числа.

**Завдання 3**

**Реалізація:**

Програмна реалізація мовою С



Код завдання 3, приклад 1



Скріншот виконання коду, приклад 1

Ця програма виконує наступні дії:

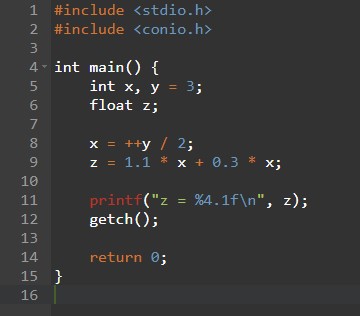
1. Ініціалізує цілочисельну змінну x зі значенням 2 та цілочисельну змінну z.
2. Ініціалізує змінну з плаваючою точкою y.
3. Обчислює y як 2.3 помножене на x.
4. Обчислює z як 0.5 \* y плюс результат постфіксного інкременту x++, поділеного на 3, помноженого на y. Оскільки x++ є постфіксним, спочатку використовується поточне значення x для обчислення, а потім x збільшується на 1.
5. Виводить значення z у форматі цілого числа за допомогою printf.
6. getch() чекає на натискання клавіші перед закриттям програми.

Важливо зазначити, що при присвоєнні значення z, відбувається неявне перетворення з float до int, що призводить до втрати дробової частини. Також, оскільки z є типу int, виведення z за допомогою %d в printf відображатиме лише цілу частину.

**Завдання 4**

**Реалізація:**

Програмна реалізація мовою С



Код завдання 4, приклад 1



Скріншот виконання коду, приклад 1

Ця програма виконує наступні дії:

1. Ініціалізує змінну y як ціле число зі значенням 3.
2. Використовує префіксний інкремент ++y, щоб збільшити y до 4, а потім ділить на 2, отже x отримує значення 2.
3. Обчислює z як суму 1.1 помноженого на x і 0.3 помноженого на x, що дає z значення 2.8.
4. Виводить z з одним десятковим знаком.
5. getch() чекає на натискання клавіші перед закриттям програми.

Таким чином, програма демонструє використання арифметичних операцій та типів даних у мові C.

# Контрольні питання

1. Призначення та структура програми, написаної мовою C.

**Призначення та структура програми, написаної мовою C**: Програма на мові C призначена для виконання певних обчислювальних завдань і складається з функцій, команд препроцесора та зовнішніх описів. [Одна з функцій має ім’я main і є точкою входу програми](https://csbc-edu.github.io/programming-essentials/slides/theme02/02_programm_structure.pdf).

1. Різновиди типів величин.

[**Різновиди типів величин**: У програмуванні існують різні типи даних, такі як числові (цілі та з плаваючою комою), логічні, символьні та рядкові](https://vseosvita.ua/library/konspekt-urok-informatiki-dla-8-klasu-velicini-zminni-i-konstanti-ihni-vlastivosti-prosti-tipi-velicin-cislovij-logicnij-simvolnij-radkovij-129964.html).

1. Що таке константи і змінні?

[**Константи і змінні**: Константа – це величина, яка не змінюється протягом виконання програми, тоді як змінна – це комірка пам’яті, яка може змінювати своє значення](https://support.almedia.com.ua/chim-vidriznyaietsya-konstanta-vid-prostoi-zminnoi/).

1. Порядок виконання операцій.

[**Порядок виконання операцій**: В програмуванні операції виконуються згідно з пріоритетом операторів та правилами асоціативності, які визначають порядок виконання операцій](https://acode.com.ua/urok-41-priorytet-operatsij-i-pravyla-asotsiatyvnosti/).

1. Особливості операцій інкремента і декремента.

**Операції інкремента і декремента**: Інкремент (++) збільшує значення змінної на одиницю, а декремент (--) зменшує його на одиницю. [Ці операції можуть бути префіксними (наприклад, ++x) або постфіксними (наприклад, x++), що впливає на порядок виконання операцій](https://codelessons.dev/ru/inkrement-i-dekrement-v-c/).

1. Операції присвоєння.

[**Операції присвоєння**: Операція присвоєння (=) встановлює значення правого операнда лівому операнду, який повинен бути змінною](https://abitap.com/2-17-operacziyi-prysvoyennya/).

1. Пояснити зміст і обгрунтувати результати виконаних прикладів.

**Пояснення результатів виконаних прикладів**: Для пояснення результатів виконаних прикладів необхідно знати контекст задачі та вхідні дані. [Загалом, результати виконаних прикладів залежать від використаних операторів та порядку їх виконання](https://foxminded.ua/operator-u-prohramuvanni/).

Висновок:

Під час виконання лабораторної роботи я ознайомився з основами синтаксису арифметичних операцій та їх пріоритетом застосування в мові програмування C. В процесі виконання я ретельно розглянув теоретичні відомості та відпрацював приклади використання різних арифметичних операцій. Надалі я складав та виконував програми, які використовували ці операції для обчислення математичних виразів.

Під час написання програми я звернув увагу на синтаксичні помилки та їх виправлення для успішної компіляції. Кожен приклад програми був виконаний та результат його роботи був представлений у звіті.

Загалом, ця лабораторна робота дала мені можливість краще зрозуміти принципи роботи арифметичних операцій в мові програмування C та навчитися їх використовувати для обчислення складних математичних виразів.