МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Кафедра КІТАМ
Дисципліна «Програмування»

Звіт

з лабораторної роботи №3
«Умовні оператори мови С++»

Виконала:
ст.гр. АКТСІз-23-1
Рацебарська А.Д.

Прийняла:
доц. Максимова С.С.,

Харків 2023

3. УМОВНІ ОПЕРАТОРИ МОВИ С++

2.1 Мета роботи:

Вивчити особливості використання умовних операторів if і switch.

2.2 Теоретичні відомості:

Умовні оператори мови С++

Умовні оператори використовуються для контролю потоку виконання програми на основі певних умов. Основні умовні оператори в С++:

if та else: Як було зазначено, ці оператори використовуються для виконання різних дій залежно від того, чи є певна умова істинною чи ні.
else if: Це використовується для перевірки декількох умов одна за одною. Якщо умова в if не виконується, то програма може перевірити додаткову умову в else if.
switch: Дозволяє виконувати різні дії на основі значення змінної. Це корисно, коли потрібно порівняти одну змінну з багатьма можливими значеннями.
Інші важливі оператори в С++:

Циклічні оператори: Це включає for, while, do-while, які використовуються для виконання блоку коду багаторазово до тих пір, поки виконується певна умова.

Оператор goto: Хоча використання goto у більшості випадків не рекомендується, оскільки воно може ускладнити читання та розуміння коду, він дозволяє перейти до певного рядка в програмі.

Оператори зворотного зв'язку (break та continue): break використовується для негайного виходу з циклу або switch-блоку. Continue перериває поточну ітерацію циклу та переходить до наступної ітерації.

Тернарний оператор (?:): Це укорочена форма if-else, яка дозволяє виконувати присвоєння або повертати значення на основі умови в одному рядку.

2.3 Хід роботи:
Варіант **2**
Написати програму на С++ для:
1. перевірки попадання введеного числа в діапазон від -2 до 2 (Оператор if);
2. перекладу введеного символу від A до F в нижній регістр (Оператор case).

***Пояснення виразу char(tolower(ch))

Вираз char(tolower(ch)) в коді C++ використовується для перетворення символу ch у нижній регістр, використовуючи функцію tolower() з бібліотеки cctype. Вказівка char перед tolower(ch) є явним приведенням типу, яке забезпечує, що результат, повернутий функцією tolower(), буде оброблено як символ типу char.

Тобто це важливо бо функція tolower() працює з цілочисельними значеннями (int), а не безпосередньо з символами (char). Коли символ подається на вхід tolower(), він автоматично перетворюється в цілочисельне значення, що відповідає його ASCII-коду.

Після того, як tolower() обробляє символ, вона повертає цілочисельний результат, який відповідає ASCII-коду символу у нижньому регістрі. Явне приведення типу за допомогою char перетворює це цілочисельне значення назад у символ типу char. Це важливо для коректного виведення символу, а не його числового ASCII-коду. Отже, використання char(tolower(ch)) гарантує, що результатом буде символ у нижньому регістрі, який можна вивести або використати далі як символьне значення.

ВИСНОВКИ

У процесі виконання цієї лабораторної роботи було засвоїла важливі аспекти використання умовних операторів в мові програмування C++. Отримала практичний досвід роботи з операторами **if**, **else**, та **switch**, що є ключовими для контролю потоку виконання програм. Навчилася використовувати оператор **if** для перевірки попадання числа в певний діапазон та оператор **switch** для обробки вводу символів і переведення їх у нижній регістр. Також засвоїла як використовувати функцію **tolower()** для перетворення символів і правильно застосовувати явне приведення типів у C++. Умовні оператори є фундаментальними у мові C++ і дозволяють писати ефективні програми, здатні приймати рішення на основі даних, введених користувачем або отриманих під час виконання програми.