МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП’ЮТЕРНИХ НАУК

Лабораторна робота № 1

на тему «Типи даних, змінні та символи»

З дисципліни:

«Спортивне програмування»

Виконала: Чорномаз В. Р., група ІТ-33

Перевірив: Прокопенко В.М.

СУМИ – 2025

**Постановка задачі**

**Завдання до лабораторної роботи**(***обов'язково для всіх)*.**

1. Поділити два цілих числа 5 та 2 і вивести результат як float. (0.2 бала)
2. Оголосити константну змінну типу float та присвоїти їй значення 3.14. Вивести її значення на екран. (0.2 бала)
3. Написати програму, яка оголошує змінну типу char та виводить її ASCII-код. (0.2 бала)
4. Користувач вводить ціле і дробове число. Виведіть їх суму. (0.2 бала)
5. Напишіть програму, яка зчитує символ та виводить наступний символ у таблиці ASCII.(0.2 бала)

**Творчі завдання (не обов'язково).**

1. Придумати тест на дану тему з одним питанням і чотирма варіантами відповідей та розписати чому саме такі варіанти. Тест повинен бути цікавим і повчальним, там повинен розглядатися якийсь нюанс роботи з мовою С. (1 бал).
2. Придумати одне завдання з даної теми і повністю його розібрати. Завдання повино бути цікавим і повчальним, в ньому повинен розглядатися якийсь нюанс роботи з мовою С. (1 бал)

**Вирішення завдання 1**

**Лістинг коду:**

#include <iostream>

using namespace std;

int main() {

    float result = 5 / 2.0;

    cout << "Result of 5 / 2: " << result << endl;

    const float pi = 3.14;

    cout << "Constant pi: " << pi << endl;

    char ch = 'A';

    cout << "ASCII code of char: " << (int)ch << endl;

    int i;

    float f;

    cout << "Enter an integer: ";

    cin >> i;

    cout << "Enter a float: ";

    cin >> f;

    cout << "Sum: " << i + f << endl;

    char inputChar;

    cout << "Enter a character: ";

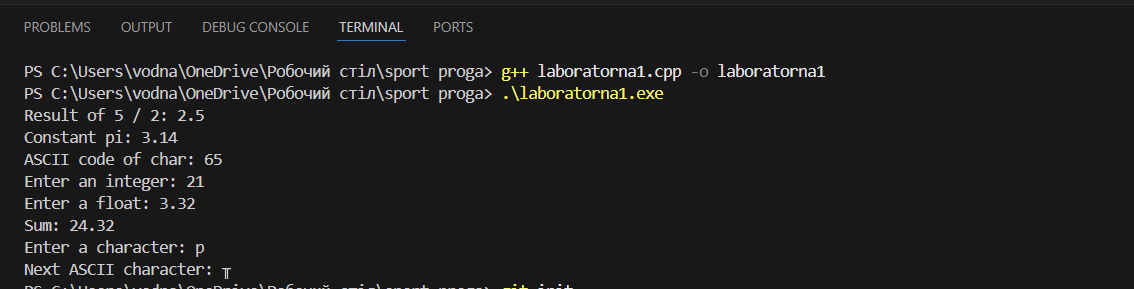
    cin >> inputChar;

    char nextChar = inputChar + 1;

    cout << "Next ASCII character: " << nextChar << endl;

    return 0;

}

****

**Рисунок 1 - Результат виконання програми №1**

**Творчі завдання**

* 1. **Що буде виведено на екран після виконання цього коду на C?**

**Лістинг коду:**

#include <stdio.h>

int main() {

    int a = 5;

    float b = 2;

    printf("%d", a / b);

    return 0;

}

**Варіанти відповідей:**

**A)** 2 **- правильна відповідь  
B)** 2.5 **C) Помилка компіляції  
D)** **Невизначена поведінка**

**Пояснення:** У цьому коді здається, що все нормально: ми ж ділимо int на float, отже результат має бути 2.5. І це справді так — в момент ділення a автоматично перетворюється на float, і математично все правильно. Але проблема виникає не там, де всі очікують — вона в printf. Там стоїть %d, який призначений для виведення цілих чисел, тобто int. А ми насправді передаємо float. І от тут починається плутанина.

Компілятор не зупиняє програму — він, скоріше за все, просто попередить (warning), але сам код може вивести щось дивне або неочікуване. Наприклад, замість 2.5 просто з'явиться 2, або взагалі якісь “сміттєві” значення. І хоча програма не падає, це вже некоректна поведінка. Це класичний приклад, як навіть маленька неуважність у форматі може призвести до неправильного результату. У C з цим треба бути дуже обережним.

**Завдання 2) Лістинг коду:**

1. #include <stdio.h>
2. int main() {
3. char ch;
4. printf("Enter a character: ");
5. scanf(" %c", &ch);
6. if (ch >= '0' && ch <= '9') {
7. int value = ch - '0';
8. printf("You entered digit: %d\n", value);
9. } else {
10. printf("Not a digit.\n");
11. }
12. return 0;
13. }

**Розбір:** у таблиці ASCII символи від '0' до '9' йдуть підряд, і мають значення від 48 до 57. Завдяки цьому можна легко перетворити будь-який символ-цифру на відповідне число — достатньо просто відняти '0'. Наприклад, '5' - '0' дає результат 5, бо в числовому представленні це 53 мінус 48.

Це працює, бо в мові C символи зберігаються як числові значення типу char, тобто фактично як маленькі цілі числа. Такий підхід є дуже ефективним і швидким, і часто використовується в різних алгоритмах. Але для новачків це може бути неочевидним моментом, оскільки ззовні ми працюємо наче з літерами, а насправді — з числами. Саме в цьому і криється цікавий нюанс мови C.

Посилання на GitHub-репозиторій (тут зберігатимуться файли з вихідним кодом до кожної лабораторної роботи) :

<https://github.com/dinoriko/sport-proga>