Міністерство освіти та науки України

Інститут «Телекомунікації, радіоелектроніки та радіотехніки»

Звіт до лабораторної роботи № 10-11

З дисципліни «Програмування частина 2»

Умовні оператори та оператори переходу у мові С

Виконав студент групи АП-12:

Дроф’як Святослав

Перевірив викладач:

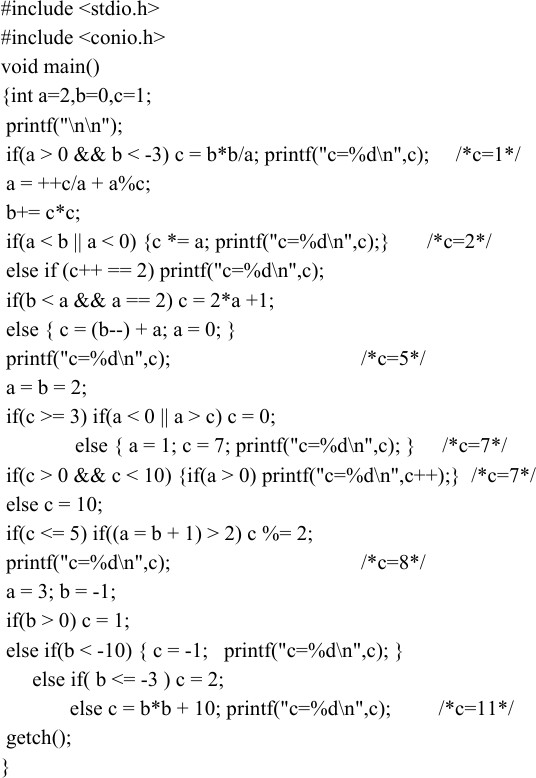
Пелех Н.В.

Львів 2024

Мета роботи: навчитися використовувати умовні оператори та оператори переходу під час програмування на мові С.

# 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. Виконати програму, яка ілюструє розгалуження умовними операторами та пояснити отримані результати:

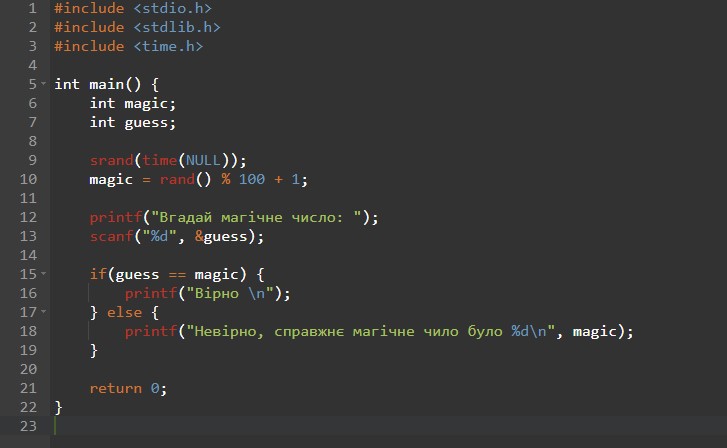


4. Написати програму для здійснення базових арифметичних операцій (додавання, віднімання, множення, ділення) над двома числами, використовуючи 13 умовний оператор if. Врахувати, що на нуль ділити неможна. Значення чисел та знак операції вводяться з клавіатури.

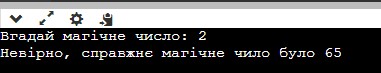
**Завдання 1**

**Реалізація:**

Програмна реалізація мовою С



Код завдання 1, приклад 1

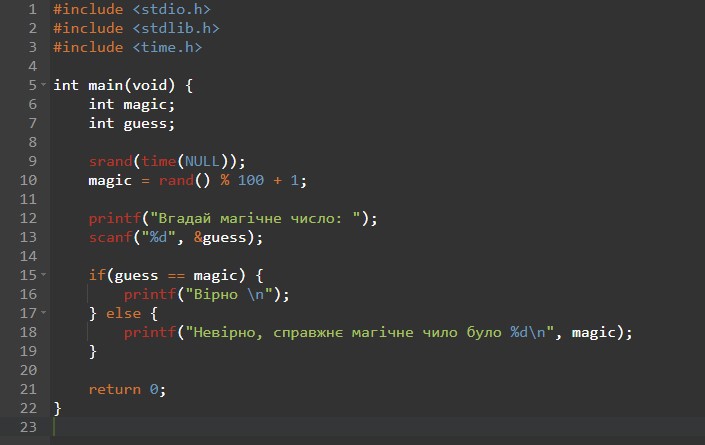


Скріншот виконання коду, приклад 1

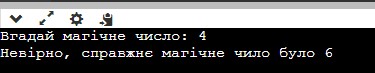
Ця програма на мові C створює просту гру на вгадування чисел. Ось її кроки:

1. #include директиви підключають стандартні бібліотеки для вводу/виводу, роботи з пам’яттю та часом.
2. Функція main() - це основна точка входу програми.
3. Змінні magic і guess оголошуються для зберігання магічного числа та числа, яке вгадує користувач.
4. srand(time(NULL)) ініціалізує генератор випадкових чисел, щоб кожен раз при запуску програми генерувались різні числа.
5. magic = rand() % 100 + 1 генерує випадкове число від 1 до 100.
6. printf виводить повідомлення, що просить користувача вгадати число.
7. scanf зчитує введене користувачем число.
8. if перевіряє, чи вгадане число співпадає з магічним. Якщо так, виводиться повідомлення “Вірно”, інакше - “Невірно” з відображенням справжнього магічного числа.
9. return 0; завершує програму з кодом успіху.

Користувач має одну спробу вгадати число. Якщо вгадує - отримує повідомлення про успіх, якщо ні - програма повідомляє правильну відповідь і завершується.



Код завдання 1, приклад 2



Скріншот виконання коду, приклад 2

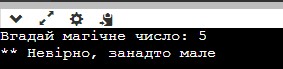
Ця програма на мові C є грою на вгадування чисел. Ось що вона робить:

1. Підключає бібліотеки для вводу/виводу та роботи з часом.
2. Генерує випадкове число від 1 до 100.
3. Запитує у користувача вгадати число.
4. Порівнює вгадане число з магічним.
5. Виводить результат: “Вірно” або “Невірно” з правильним числом.

Програма дає користувачу одну спробу вгадати магічне число.



Код завдання 1, приклад 3

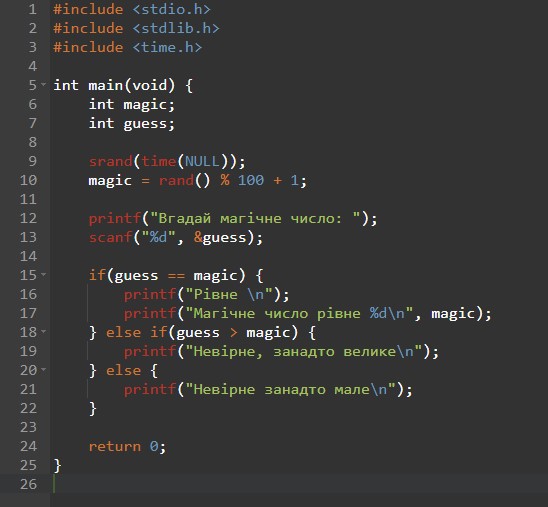


Скріншот виконання коду, приклад 3

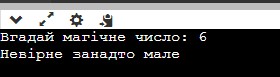
Ця програма є грою на вгадування чисел з додатковими підказками. Ось її функції:

* Ініціалізує генератор випадкових чисел.
* Генерує випадкове число від 1 до 100.
* Запитує користувача вгадати число.
* Перевіряє, чи вгадане число є магічним.
* Якщо вгадано, виводить “Вірно” та показує магічне число.
* Якщо не вгадано, виводить “Невірно” та підказку, чи число було занадто велике чи мале.

Програма надає користувачу інформацію для кращого вгадування на наступну спробу.

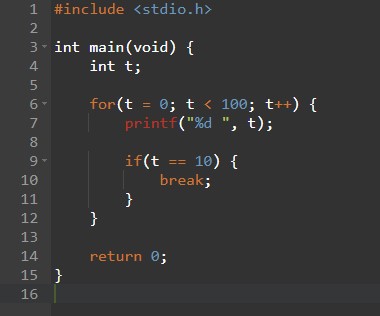


Код завдання 1, приклад 4



Скріншот виконання коду, приклад 4

Ця програма - гра на вгадування числа. Вона генерує випадкове число від 1 до 100 і просить користувача його вгадати. Якщо вгадане число співпадає з генерованим, програма виводить “Рівне” та показує це число. Якщо вгадане число більше або менше генерованого, програма вказує на це, виводячи “Невірне, занадто велике” або “Невірне, занадто мале”.



Код завдання 1, приклад 5



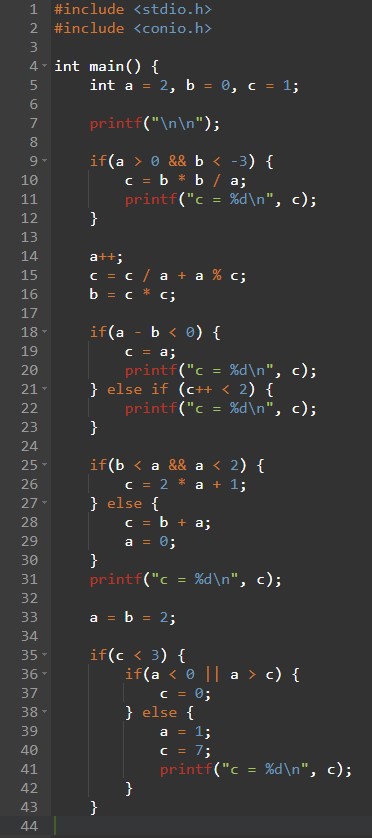
Скріншот виконання коду, приклад 5

Ця програма виводить числа від 0 до 10. Вона використовує цикл for для друкування кожного числа в рядку, але перериває цикл, коли t досягає 10 за допомогою оператора break. Таким чином, числа після 10 не друкуються.

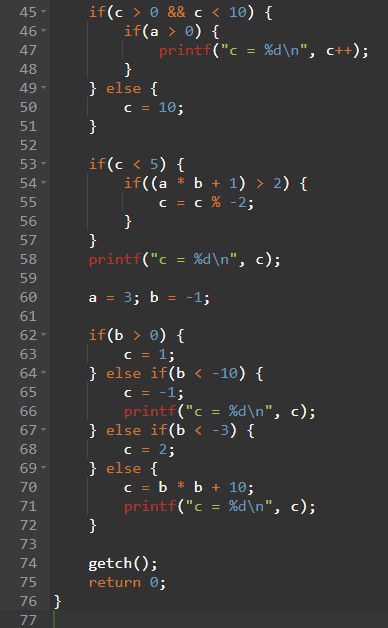
**Завдання 2**

**Реалізація:**

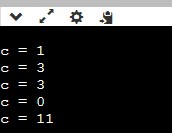
Програмна реалізація мовою С



Код завдання 2



Код завдання 2



Скріншот виконання коду, завдання 2

Ця програма виконує ряд математичних операцій та умовних перевірок з трьома змінними a, b, і c. Ось короткий опис її дій:

1. Ініціалізує змінні a, b, і c.
2. Виконує умовні перевірки та обчислення, змінюючи значення c.
3. Виводить значення c після кожної зміни.
4. Використовує getch() для очікування натискання клавіші перед завершенням.

Програма демонструє використання умовних операторів if та else, арифметичних операцій, інкрементації та декрементації змінних, і логічних умов. Вона також показує, як можна змінювати потік виконання програми залежно від різних умов.

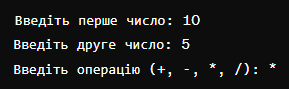
**Завдання 3**

**Реалізація:**

Програмна реалізація мовою С



Код завдання 3



Скріншот виконання коду, завдання 3



Скріншот виконання коду, завдання 3

**Контрольні питання:**

1, Умовні оператори у мові C:

if: Виконує певні дії, якщо задана умова істинна.

else: Виконується, якщо попередня умова if була хибною.

else if: Дозволяє перевірити додаткові умови, якщо попередня умова if була хибною.

switch: Використовується для перевірки різних значень змінної та виконання коду в залежності від цих значень.

2. Оператори переходу у мові C:

break: Припиняє виконання циклу або switch-блоку, і контроль передається наступному оператору поза циклом або switch-блоком.

continue: Припускає поточну ітерацію циклу та переходить до наступної ітерації циклу.

goto: Використовується для безумовного переходу до мітки в програмі, але використання його не рекомендується через можливість порушення логіки програми.

3. Умовний оператор if:

if виконує певний фрагмент коду, якщо задана умова є істинною. У разі, якщо умова не виконується, код, який знаходиться після блоку if, пропускається або виконується блок else, якщо він присутній.

4. Оператор break:

break призначений для припинення виконання найближчого внутрішнього циклу або switch-блоку. Використовується для виходу з циклу, якщо досягнуто певної умови або завершення обробки в switch-блоці.

5.Функція rand:

rand генерує випадкове ціле число від 0 до RAND\_MAX (макрос, що зазвичай дорівнює 32767). Щоб використовувати rand, спочатку потрібно ініціалізувати генератор псевдовипадкових чисел за допомогою функції srand. Кожного разу, коли ви викликаєте rand, він повертає нове випадкове число.

**Висновок:**

Під час цієї лабораторної роботи я поглибив знання про умовні оператори та оператори переходу в мові програмування C. Я створив програми для обчислення виразу та для виконання базових арифметичних операцій над двома числами з використанням цих конструкцій. Ці завдання допомогли мені краще зрозуміти та впровадити умовні конструкції у практичних задачах програмування.