Міністерство освіти і науки України

Національний університет “Львівська Політехніка”

Лабораторна робота №10-11

З дисципліни

“Програмування частина 2”

Виконав:

Студент групи АП-11

Гишка Остап

Прийняв:

Чайковський І.Б.

Львів-2024

«Умовні оператори та оператори переходу у мові С»

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

Теоретичні відомості

Умовні оператори та оператори переходу - це важливий елемент у будь-якій мові програмування, включаючи мову С. Вони дозволяють контролювати хід виконання програми в залежності від певних умов, що виконуються під час її виконання. Ось короткий огляд цих конструкцій:

Умовні оператори:

if-else: Ця конструкція дозволяє виконувати певні дії, якщо вираз умови є істинним, або інші дії, якщо він є хибним.

if (условие) {

// Виконується, якщо умова істинна

} else {

// Виконується, якщо умова хибна

}

else-if: Можна використовувати, коли потрібно перевірити кілька умов послідовно.

if (условие1) {

// Виконується, якщо умова1 істинна

} else if (условие2) {

// Виконується, якщо умова2 істинна

} else {

// Виконується, якщо ні одна з умов не виконується

}

Оператори переходу:

break: Використовується для виходу з циклу або перемикача.

continue: Переходить до наступної ітерації циклу.

return: Повертає значення з функції та завершує її виконання.

goto: Використовується для безумовного переходу до певної мітки в програмі, але його використання не рекомендується через можливість створення складних структур і важкості відлагодження.

Ці конструкції дозволяють створювати більш гнучкі програми, які можуть виконувати різні дії в залежності від умов, а також контролювати потік виконання програми з використанням операторів переходу. Однак, слід ретельно використовувати оператори переходу, оскільки вони можуть призвести до складного та неочевидного коду.

**Приклад 1**

/\* Магічне число \*/

#include <stdio.h>

#include <stdlib.h>

main()

{

int magic; /\* магічне число \*/

int guess; /\* спроба гравця \*/

magic = rand(); /\* генерація магічного числа \*/

printf("Vgaday magichne chuslo: ");

scanf("%d", &guess);

if(guess == magic) printf("\*\* Virno \*\*");

return 0;

}



**Приклад 2**

/\*Магічне число\*/

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

int main()

{

int magic; //магічне число

int guess; //спроба гравця

//magic=rand(0,10);

srand(time(NULL));

magic=rand()%10; //генерація маг число

printf("Ваша здогадка: ");

scanf("%d", &guess);

if(guess==magic)printf("\*\*Virno\*\*\n");

else printf("Nevirno");

return 0;

}



**Приклад 3**

#include <stdio.h>

#include <stdlib.h>

#include <time.h> // Підключаємо бібліотеку для використання функції time()

int main(void) {

int magic;

int guess;

// Ініціалізуємо генератор випадкових чисел за поточним часом

srand(time(NULL));

// Генеруємо магічне число в діапазоні від 0 до 9

magic = rand() % 10;

printf("Вгадайте магічне число: ");

scanf("%d", &guess);

if (guess == magic) {

printf("\*\* Вірно \*\*\n");

printf("Магічне число рівне %d\n", magic);

} else {

printf("\*\* Невірно, ");

if(guess > magic)

printf("занадто велике\n");

else

printf("занадто мале\n");

}

 return 0;

}

**Приклад 4**

/\*Магічне число 4\*/

#include <stdio.h>

#include <stdlib.h>

int main(void) {

int magic;

int guess;

// Генеруємо випадкове магічне число в діапазоні від 0 до RAND\_MAX

magic = rand() % 10;

printf("Вгадайте магічне число: ");

scanf("%d", &guess);

if (guess == magic) {

printf("\*\*\* Вірно \*\*\*\n");

printf("Магічне число рівне %d\n", magic);

} else if (guess > magic) {

printf("Невірно, занадто велике\n");

} else {

printf("Невірно, занадто мале\n");

}

return 0;

}



**Приклад 5**

#include <stdio.h>

int main(void)

{

int t;

for(t=0; t<100; t++) {

printf("%d ", t);

if(t==10) break;

}

return 0;

}



**Приклад 6**

/\* Підрахунок кількості пробілів \*/

#include <stdio.h>

int main(void)

{

char s[80], \*str;

int space;

printf("Vvedit riadok: ");

gets(s);

str = s;

for(space=0; \*str; str++) {

if(\*str != ' ') continue;

space++;

}

printf("%d probiliv\n", space);

return 0;

}

**Приклад 7**

#include <stdio.h>

#include <conio.h> // Підключення для getch()

void main() {

int a = 2, b = 0, c = 1;

printf("\n\n");

if (a > 0 && b < -3) // Виправлення: Додано пропущені фігурні дужки

c = b \* b / a;

printf("c=%d\n", c); // c=1

a = ++c / a + a % c; // Виправлення: Змінено ++c на ++c / a для кращої читабельності

b = +c \* c; // Виправлення: Видалено зайвий + перед c

if (a < b || a < 0) { // Виправлення: Додано пропущені фігурні дужки та замінено || на |

c \*= a;

printf("c=%d\n", c); // c=2

} else if (c++ == 2) // Виправлення: Замінено = на == для порівняння

printf("c=%d\n", c); // c=2

if (b < a && a == 2) // Виправлення: Додано пропущені фігурні дужки

c = 2 \* a + 1;

else {

c = (b--) + a;

a = 0;

}

printf("c=%d\n", c); // c=5

a = b = 2;

if (c >= 3) {

if (a < 0 || a > c) // Виправлення: Додано пропущені фігурні дужки

c = 0;

else {

a = 1;

c = 7;

printf("c=%d\n", c); // c=7

if (c > 0 && c < 10) {

if (a > 0)

printf("c=%d\n", c++); // c=7

}

else

c = 10;

}

}

if (c <= 5) {

if ((a = b + 1) > 2) // Виправлення: Додано пропущені фігурні дужки

c % 2;

}

printf("c=%d\n", c); // c=8

a = 3;

b = -3;

if (b > 0)

c = 1;

else if (b < -10) {

c = -1;

printf("c=%d\n", c); // c=-1

}

else if (b <= -3)

c = 2;

else

c = b \* b + 10;

printf("c=%d\n", c); // c=11

getch(); // Виклик getch() для очікування натискання клавіші

}



**Приклад 8**

#include <stdio.h>

int main() {

float num1, num2;

char operator;

printf("Введіть перше число: ");

scanf("%f", &num1);

printf("Введіть друге число: ");

scanf("%f", &num2);

printf("Введіть операцію (+, -, \*, /): ");

scanf(" %c", &operator);

if (operator == '+') {

printf("%.2f + %.2f = %.2f\n", num1, num2, num1 + num2);

} else if (operator == '-') {

printf("%.2f - %.2f = %.2f\n", num1, num2, num1 - num2);

} else if (operator == '\*') {

printf("%.2f \* %.2f = %.2f\n", num1, num2, num1 \* num2);

} else if (operator == '/') {

if (num2 != 0) {

printf("%.2f / %.2f = %.2f\n", num1, num2, num1 / num2);

} else {

printf("Помилка: ділення на нуль\n");

}

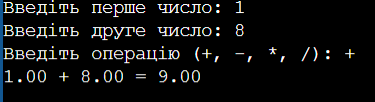
} else {

printf("Помилка: невідома операція\n");

}

return 0;

}



Відповіді на контрольні запитання

1)Назвіть умовні оператори у мові С:

У мові програмування С існують наступні умовні оператори:

if: Використовується для виконання певної дії, якщо умова істинна.

else: Використовується разом з оператором if для виконання іншої дії, якщо умова оператора if хибна.

else if: Використовується для перевірки додаткових умов після if.

switch: Використовується для вибору виконання одного зі списку можливих варіантів.

2)Назвіть оператори переходу у мові С:

Оператори переходу в мові С включають:

break: Використовується для виходу з циклу або перемикача.

return: Використовується для повернення значення з функції та завершення її виконання.

3) Охарактеризуйте умовний оператор if:

Оператор if використовується для виконання певної дії, якщо вираз умови істинний. Він має наступну структуру:

if (вираз\_умови) {

// Виконується, якщо вираз\_умови істинний

}

4)Охарактеризуйте призначення оператора break: Оператор break використовується для виходу з циклу або перемикача. Він перериває виконання циклу або перемикача, який його містить, і переходить до наступного оператора після циклу або перемикача.

5)Охарактеризуйте призначення оператора break: Оператор break використовується для виходу з циклу або перемикача. Він перериває виконання циклу або перемикача, який його містить, і переходить до наступного оператора після циклу або перемикача.