

Тема 3. Умовні конструкції (if, else, switch).

У мові програмування C умовні конструкції дозволяють виконувати різні блоки коду залежно від виконання певних умов. Основними умовними конструкціями є if, else та switch.

1. Конструкція if

Конструкція if використовується для перевірки умови, і якщо умова істинна (тобто рівна 1), то виконується блок коду, що йде після неї.

Синтаксис:

```
if (умова) {  
    // Блок коду, що виконується, якщо умова істинна  
}
```

Приклад:

```
int a = 5;  
if (a > 0) {  
    printf("a більше за 0\n");  
}
```

Якщо умова в дужках після if істинна, то буде виконано тіло конструкції (виведення "a більше за 0").

2. Конструкція else

Конструкція else використовується для виконання блоку коду, коли умова в конструкції if не виконується (тобто умова хибна).

Синтаксис:

```
if (умова) {  
    // Блок коду, що виконується, якщо умова істинна  
} else {  
    // Блок коду, що виконується, якщо умова хибна  
}
```

Приклад:

```
int a = -5;  
if (a > 0) {  
    printf("a більше за 0\n");  
} else {  
    printf("a не більше за 0\n");  
}
```

У цьому випадку, оскільки `a` не більше за 0, виведеться "а не більше за 0".

3. Конструкція `else if`

Конструкція `else if` дозволяє додати кілька умов, щоб перевірити різні варіанти.

Синтаксис:

```
if (умова1) {  
    // Блок коду, що виконується, якщо умова1 істинна  
} else if (умова2) {  
    // Блок коду, що виконується, якщо умова2 істинна  
} else {  
    // Блок коду, що виконується, якщо жодна умова не істинна  
}
```

Приклад:

```
int a = 0;  
if (a > 0) {  
    printf("а більше за 0\n");  
} else if (a < 0) {  
    printf("а менше за 0\n");  
} else {  
    printf("а дорівнює 0\n");  
}
```

У цьому прикладі виведеться "а дорівнює 0", оскільки умова `a == 0` відповідає істині.

4. Конструкція `switch`

Конструкція `switch` використовується для перевірки одного значення на відповідність кільком варіантам. Це зручніше, коли потрібно порівняти одну змінну з кількома можливими значеннями.

Синтаксис:

```
switch (вираз) {  
    case значення1:  
        // Код, який виконується, якщо вираз дорівнює значення1  
        break;  
    case значення2:  
        // Код, який виконується, якщо вираз дорівнює значення2  
        break;  
    default:  
        // Код, який виконується, якщо жоден case не відповідає  
}
```

- **case** — це значення, з яким порівнюється вираз. Якщо воно дорівнює значенню виразу, виконується відповідний блок коду.
- **break** — використовується для виходу з конструкції `switch`, щоб не виконувалися наступні варіанти.
- **default** — необов'язковий блок, який виконується, якщо жоден з варіантів не співпав.

Приклад:

```
int day = 3;
switch (day) {
    case 1:
        printf("Понеділок\n");
        break;
    case 2:
        printf("Вівторок\n");
        break;
    case 3:
        printf("Середа\n");
        break;
    default:
        printf("Невідомий день\n");
}
```

У цьому випадку виведеться "Середа", оскільки значення змінної `day` дорівнює 3.

Підсумок

- **if** — перевіряє одну умову.
- **else if** — дозволяє перевірити кілька умов по черзі.
- **else** — виконується, якщо жодна з умов не виконана.
- **switch** — дозволяє вибір між кількома варіантами для одного виразу, зручно, коли є багато умов для одного значення.

Ці конструкції дозволяють створювати гнучкі і потужні умови для виконання різних частин коду залежно від результатів перевірок.