

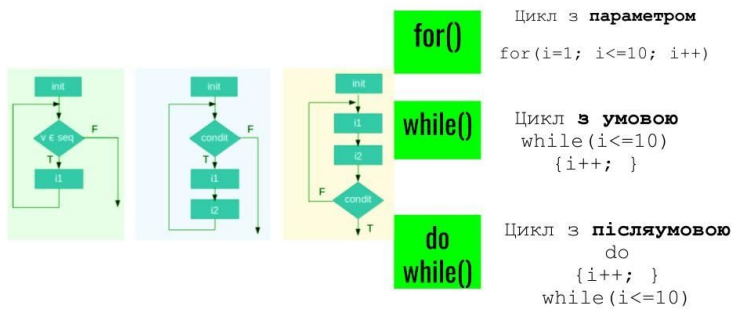


Тема 4 Цикли

Цикл - це структура програмування, яка повторює послідовність вказівок доти, доки не буде дотримана певна умова. Програмісти використовують цикли для перегляду значень, додавання сум чисел, повторення функцій та багатьох інших речей.

Цикли підтримуються всіма сучасними мовами програмування, хоча їх реалізації та синтаксис можуть відрізнятися. Два найпоширеніші типи циклів - цикл **while** та цикл **for**.

Цикли



Цикл з умовою

while()

```
{  
    Повторення дії або блоку дій  
    поки умова істинна  
}
```

```
int i=1;  
while(i<=3)  
{  
    i++;  
    cout<<i<<" ";  
}
```

Цикл з параметром

for (;){

```
    Повторення дії або блоку дій  
    певну кількість разів
```

```
}  
int i;  
for(i=1;i<=10;i++)  
{  
    cout<<i<<" "<<i*i<<endl;  
}
```

Введення та виведення 10 цілих чисел

```
#include <iostream>  
using namespace std;  
int main()  
{
```

```
int i, a;
for(i=1;i<=10;i++)
{
    cin>>a;
    cout<<a<<endl;
}
return 0;}
```

Додаємо умови в цикли

```
#include <iostream>
using namespace std;
int main() {
    int a;
    int k = 0;
    for (int i = 0; i < 10; i++)
    {
        cin >> a;
        if (a > 100) {
            k++;
        }
    }
    cout << k << endl;
    return 0;}
```

Приклад реалізації задачі знаходження суми 10 введених чисел з різними циклами:

```
#include <iostream>
using namespace std;
int main() {
    int a, i=1, sum = 0;
    while ( i<=10) {
        cin >> a;
        sum = sum + a;
        i++;
    }
    cout << "Sum = " << sum << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;
int main() {
    int a;
    int sum = 0;
    for (int i = 0; i < 10; i++) {
        cin >> a;
        sum = sum + a;
    }
    cout << "Sum = " << sum << endl;
    return 0;
}
```

Цикл з післяумовою

Цикл `do while` перевіряє умову після виконання блоку дій.

```
#include <iostream>
using namespace std;
int main() {
    int a, i=1, sum = 0;
    do
    {
        cin >> a;
        sum = sum + a;
        i++;
    }
    while ( i<=10)
    cout << sum << endl;
```

```
    return 0;
}
```

Оператори керування циклами

Break завершує виконання циклу. Управління передається оператору, який розташований після циклу.

```
#include <iostream>
using namespace std;
int main() {
    for (int i = 1; i <= 5; i++) {

        if (i == 3) {
            break;
        }
        cout << i << endl;
    }
    return 0;}
```

Continue дозволяє відразу перейти в кінець тіла циклу, пропускаючи весь код, який знаходиться під ним.

Це корисно в тих випадках, коли ми хочемо завершити поточну ітерацію завчасно.

```
#include <iostream>
using namespace std;
int main() {
    for (int i = 1; i <= 5; i++) {

        if (i == 3) {
            continue;
        }
        cout << i << endl;
    }
    return 0; }
```

Вкладені цикли

Цикл називається вкладеним, якщо він розміщується всередині іншого циклу.

```
#include <iostream>
using namespace std;
int main() {
    for(int i=1;i<9;i++)
    {cout<<endl;
        for(int j=1;j<9;j++)
            cout<<i<<"*"<<j<<"="<<i*j;
    }
    return 0;}
```