

## Лабораторна робота № 6

### ЦИКЛИ.

*Мета роботи:*

Вивчити поняття і застосування циклів.

*Обладнання та програмне забезпечення:*

- IBM сумісна персональна обчислювальна машина;
- онлайн компілятор мови програмування Cі, доступний за посиланням <https://repl.it/languages/c>

*Завдання до роботи:*

Написати програму на мові програмування Cі, котра реалізує наступні функції:

- використання оператора `for`;
- використання оператора `while`;
- використання оператора `do/while`;
- використання оператора `break`;
- використання вкладених операторів `continue`;

### Теоретичні відомості

#### Цикл `for`

`for` (ініціалізація; умова; модифікація) оператор; умова перевіряється на початку ітерації (цикл з передумовою). Цикл виконується, якщо умова істинна ( $\neq 0$ ).

```
for (int i = 0; i < 10; i++)
{
    printf("%d; ",i); // 0; 1; 2; 3; 4; 5; 6; 7; 8; 9;
}
```

#### Використання оператора об'єднання “,” :

```
for (int i = 0, j = 10; i < 10; i++, j--)
{
    printf("i= %d, j= %d\n", i, j);
}
```

#### Ініціалізація параметрів циклу до тіла циклу:

```
int i = 0, j = 10;
for (; i < 10; i++, j--)
{
    printf("i= %d, j= %d\n", i, j);
}
```

### Зміна параметрів в тілі циклу

```
int i = 0, j = 10;
for (; i < 10;)
{
    i++;
    j--;
    printf("i= %d, j= %d\n", i, j);
}
```

### Нескінченний цикл **for**

```
for ( ; ; ) printf(" this loop will run forever. \n");
```

### Цикл **for** без тіла

```
for (t=0; t<10000; t++); /* реалізація затримки часу */
```

### Цикл **while**

**while** (умова) оператор;

**умова** перевіряється на початку ітерації (цикл з передумовою). Цикл виконується, якщо умова істинна ( $\neq 0$ ).

/\* Процедура обробляє ввід з клавіатури, доки не буде введено символ A\*/

```
void wait_for_char(void)
{
    char ch='\0';
    while (ch!='A') ch = getchar();
}
```

### Цикл **do/while**

```
do
{
    послідовність операторів;
}
while (умова);
```

**умова** перевіряється в кінці ітерації (цикл з післяумовою), тому виконується принаймні одна ітерація. Якщо **умова** стає істинною, виконання циклу завершується.

```
int num;
do
{
    printf ("num= ");
    scanf ("%d", &num);
} while (num > 32);
```

## Оператор **break**

Оператор **break** має два призначення:

1. закінчення роботи оператора **switch**;
2. примусове завершення циклу без стандартної перевірки умови;

```
#include <stdio.h>
int main(void)
{
    for(int i = 0; i < 20; i++)
    {
        printf ("i= %d\n", i);
        if (i == 10) break; /* коли i досягає 10, цикл переривається */
    }
    return 0;
}
```

```
#include <stdio.h>
char ch;
int main (void)
{
    for ( ; ; )
    {
        ch = getchar(); /* ввід символу */
        if (ch == 'A') break; /* вихід з циклу */
    }
    printf("you typed an A");
    return 0;
}
```

## Оператор **continue**

Оператор **continue** переходить до наступної ітерації циклу, пропускаючи код, що залишився. В циклах **while** та **do/while** оператор **continue** викликає перехід до перевірки умови і подальшого продовження роботи циклу. Для циклу **for** виконується модифікація ітератора, потім перевіряється умова, і накінець, виконується тіло циклу.

```
#include <stdio.h>
int main(void){
    for(int i = 0; i < 20; i++){
        if (i == 10) continue; /* значення i=10 не виводиться */
        printf ("i= %d\n", i);
    }
    return 0;
}
```

```

#include <stdio.h>
int main(void)
{
    int x;
    for(int i = 0; i <= 21; i++)
    {
        scanf("%d", x);
        if (x < 0) continue; /* від'ємні значення не виводяться */
        printf ("%d ", x);
        if (x == 13) break; /* вихід з циклу */
    }
    return 0;
}

```

#### ПОРЯДОК ВИКОНАННЯ РОБОТИ:

1. Опрацювати і засвоїти матеріал наведений в теоретичних відомостях.
2. Написати програму, котра дозволяє ввести речення і підрахувати кількість голосних і приголосних у ньому.
3. Написати програму, котра в циклі **while** вираховує суму перших 100-та чисел за винятком Вашого порядкового номера в журналі і числа 33.
4. Написати програму, котра дозволяє користувачу задати ціле число  $N \in [7; 12]$ , далі в циклі ввести  $N$  дійсних чисел, розрахувати і вивести суму введених чисел, середнє арифметичне, мінімальне і максимальне значення.
5. В звіті навести копії екранів та написаний код.
6. Зробити висновки.