

Лабораторна робота № 8

Програмування циклічних алгоритмів Оператори циклу for і while

Мета: вивчити особливості використання операторів циклу.

Хід роботи: Всі завдання виконуються у двох варіантах з оператором циклу for та while.

Завдання 1 . Запишіть логічний вираз, який приймає значення "істина" тоді і тільки тоді, коли точка з координатами (x, y) належить заштрихованій області.

10	Ви поклали в банк 1500 грн. Визначте, скільки грошей буде на Вашому рахунку через 1 рік, якщо кожен місяць сума на рахунку збільшується на 0,76% від суми попередня місяця.
----	---

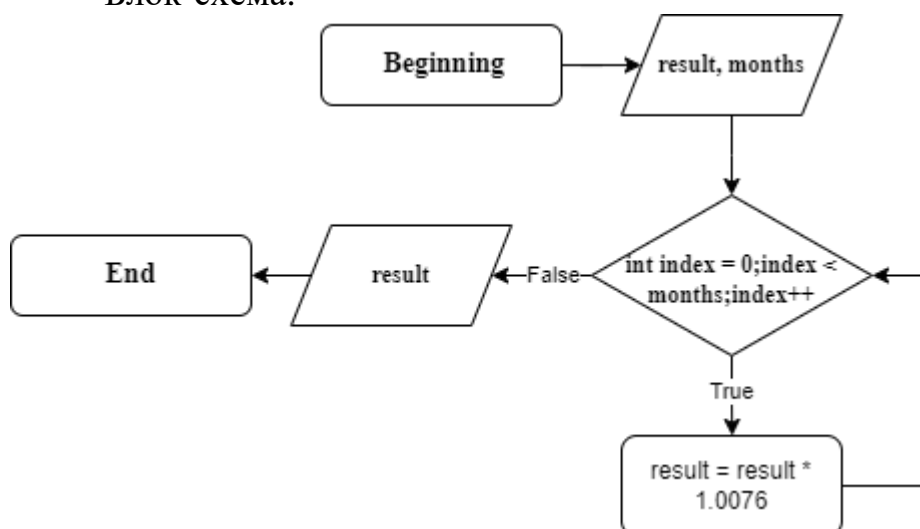
Листинг програми (for):

```
#include <stdio.h>
int main(){
    double result = 1500;
    int months = 12;
    for (int index = 0; index < months; index++) {
        result = result * 1.0076;
    }
    printf("The result in one year is %.2lf", result);
    return 0;
}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>for.exe
The result in one year is : 1642.67
```

Блок-схема:



					ДУ «Житомирська політехніка».23.121.10.000 – Лр8						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Семенчук О.А.			Звіт з лабораторної роботи			Лім.	Арк.	Аркушів	
Перевір.		Чижмотря О. В							1	11	
Керівник								ФІКТ Гр. ІПЗ-21-1[2]			
Н. контр.											
Зав. каф.											

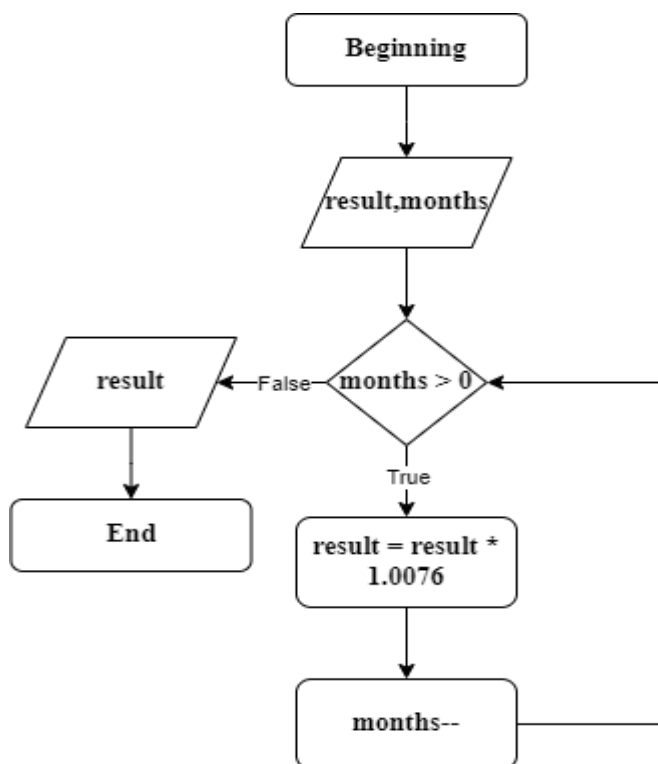
Листинг програми (while):

```
#include <stdio.h>
int main() {
    double result = 1500;
    int months = 12;
    while (months > 0) {
        result = result * 1.0076;
        months--;
    }
    printf("The result in one year is : %.2lf\n", result);
    return 0;}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>while.exe
The result in one year is : 1642.67
```

Блок-схема:



Завдання 2. Дано натуральне число n . Підрахувати:

10	$\left(1 + \frac{1}{1^2}\right) + \left(1 + \frac{1}{2^2}\right) + \dots + \left(1 + \frac{1}{n^2}\right)$
----	--

Листинг програми (for):

```
#include <stdio.h>
#include <math.h>
int main() {
    int n;
    printf("What is the value of n : "); scanf_s("%d", &n);
    double expression = 0.0;
    for (int index = 1; index <= n; index++) {
        expression = expression + (1 + 1.0 / pow(index, 2));
    }
    printf("%.2lf\n", expression);
    return 0;}
```

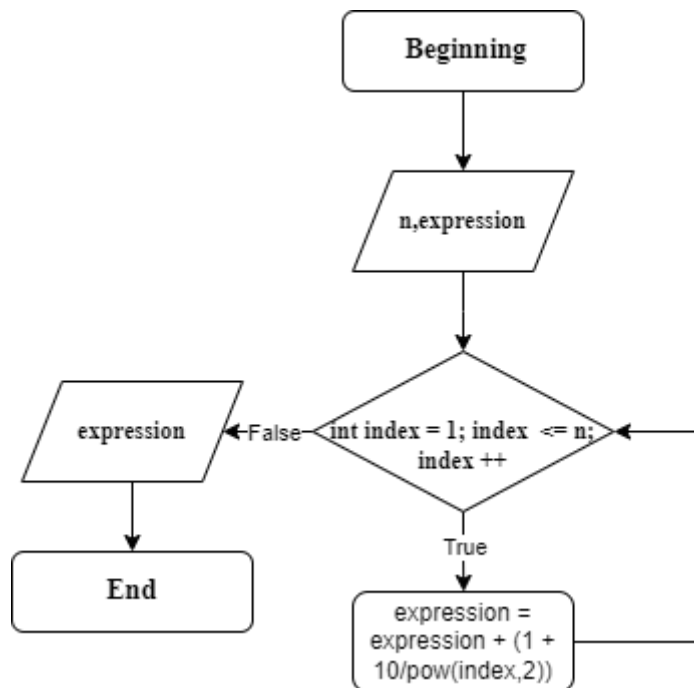
Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>for.exe
What is the value of n : 1
2.00

C:\Users\FearlessAtom\Desktop>for.exe
What is the value of n : 2
3.25

C:\Users\FearlessAtom\Desktop>for.exe
What is the value of n : 3
4.36
```

Блок-схема:



Листинг програми (**while**):

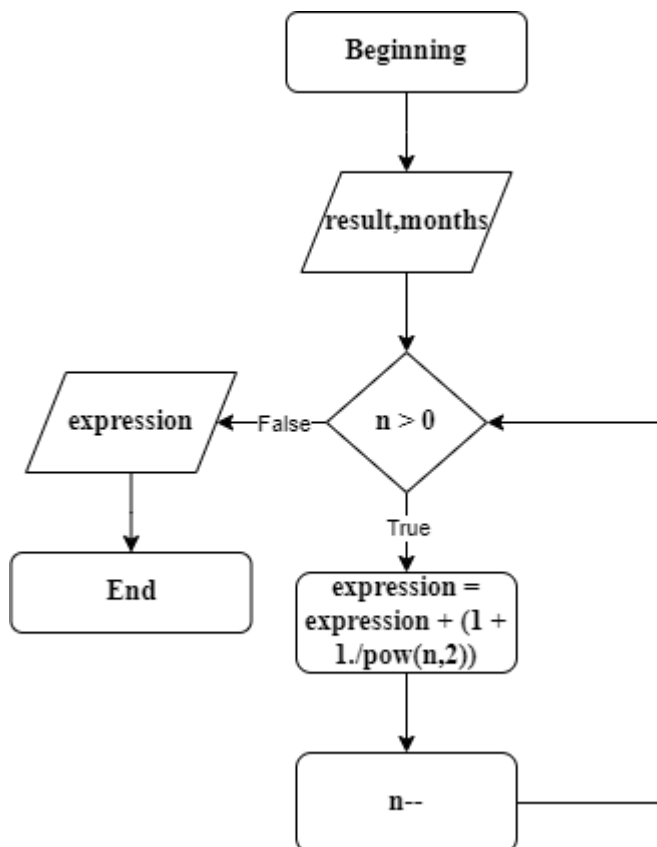
```
#include <stdio.h>
#include <math.h>
int main() {
    int n;
    printf("What is the value for n : "); scanf_s("%d", &n);
    double expression = 0.0;
    while (n > 0){
        expression = expression + (1 + 1. / pow(n, 2));
        n--;
    }
    printf("%.2lf", expression);
    return 0;}
```

Результат виконання програми:

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр8	Арк.
		Чижмоторя О. В.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

```
C:\Users\FearlessAtom\Desktop>while.exe
What is the value for n : 1
2.00
C:\Users\FearlessAtom\Desktop>while.exe
What is the value for n : 2
3.25
C:\Users\FearlessAtom\Desktop>while.exe
What is the value for n : 3
4.36
```

Блок-схема:



Завдання 3. Обчислити значення функції $F(x)$ на відрізку $[a,b]$ кроком h :

10	$(2x + 1) * \sqrt{x}$	7;8	1	$\sqrt[5]{x} \cos(x)$	1;1.5	0.1
----	-----------------------	-----	---	-----------------------	-------	-----

Листинг програми (for):

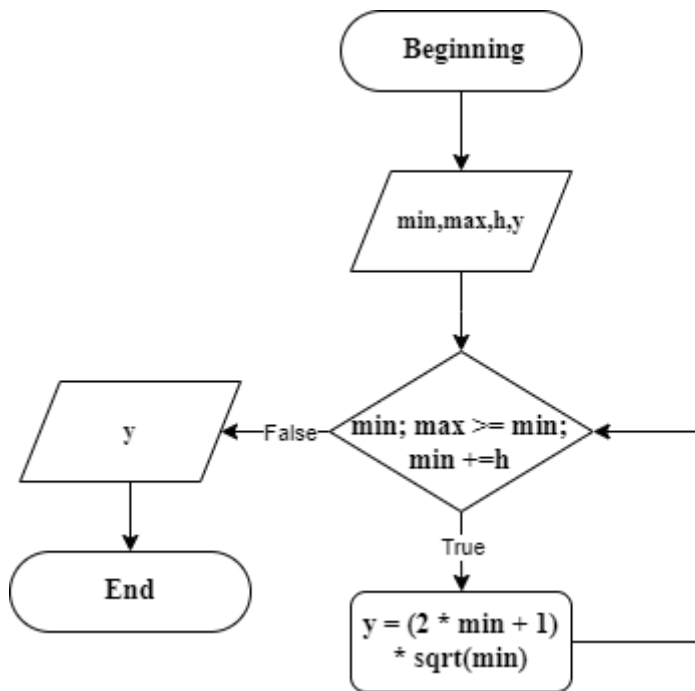
```
#include <stdio.h>
#include <math.h>
int main() {
    int min = 7, max = 8;
    double h = 1, y;
    for (min; max >= min; min += h) {
        y = (2 * min + 1) * sqrt(min);
        printf("%.2lf\n", y);
    }
    return 0;
}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>for.exe
39.69
48.08
```

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр8	Арк.
		Чижмоторя О. В.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

Блок-схема:



Листинг програми (**while**):

```

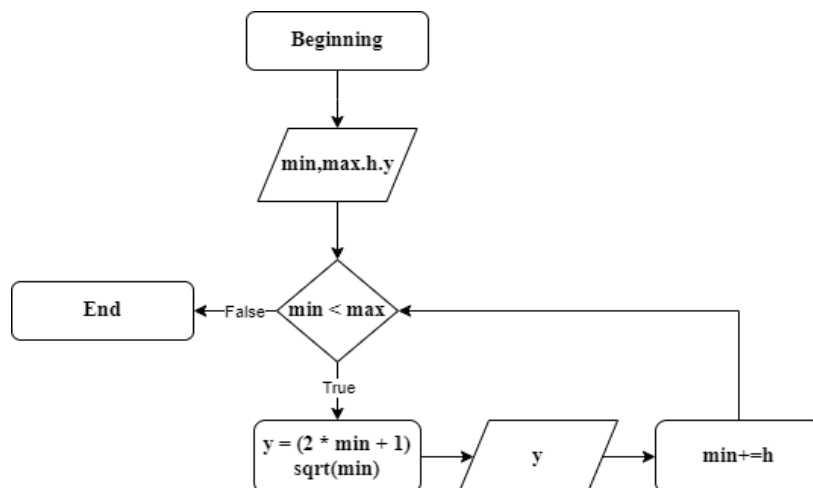
#include <stdio.h>
#include <math.h>
int main(){
    int min = 7, max = 8;
    double h = 1, y;
    while (min <= max) {
        y = (2 * min + 1) * sqrt(min);
        printf("%.2lf\n", y);
        min = min + h;}
    return 0;}
  
```

Результат виконання програми:

```

C:\Users\FearlessAtom\Desktop>while.exe
39.69
48.08
  
```

Блок-схема:



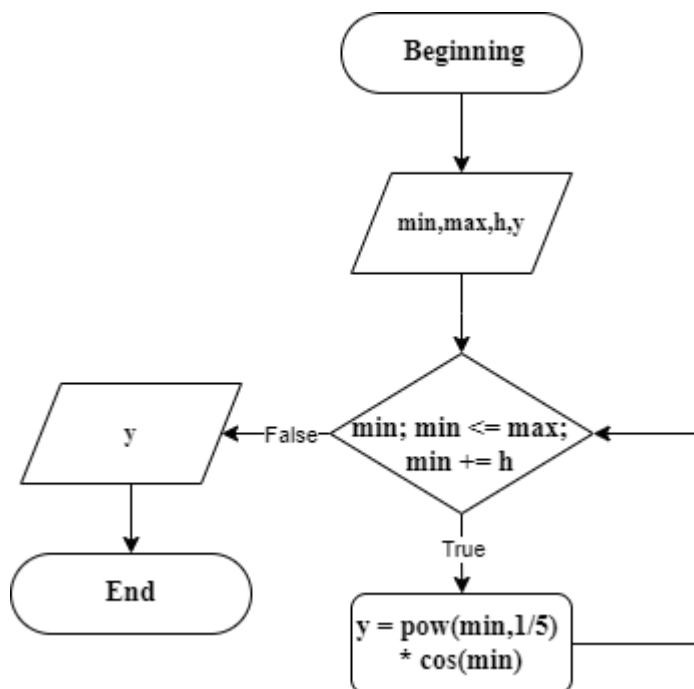
Листинг програми (for):

```
#include <stdio.h>
#include <math.h>
int main(){
    double h = 0.1, y, min = 1, max = 1.5;;
    for (min; min <= max; min += h) {
        y = pow(min, 1 / 5) * cos(min);
        printf("%.2lf\n", y);}
    return 0;}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>for.exe
0.54
0.45
0.36
0.27
0.17
```

Блок-схема:



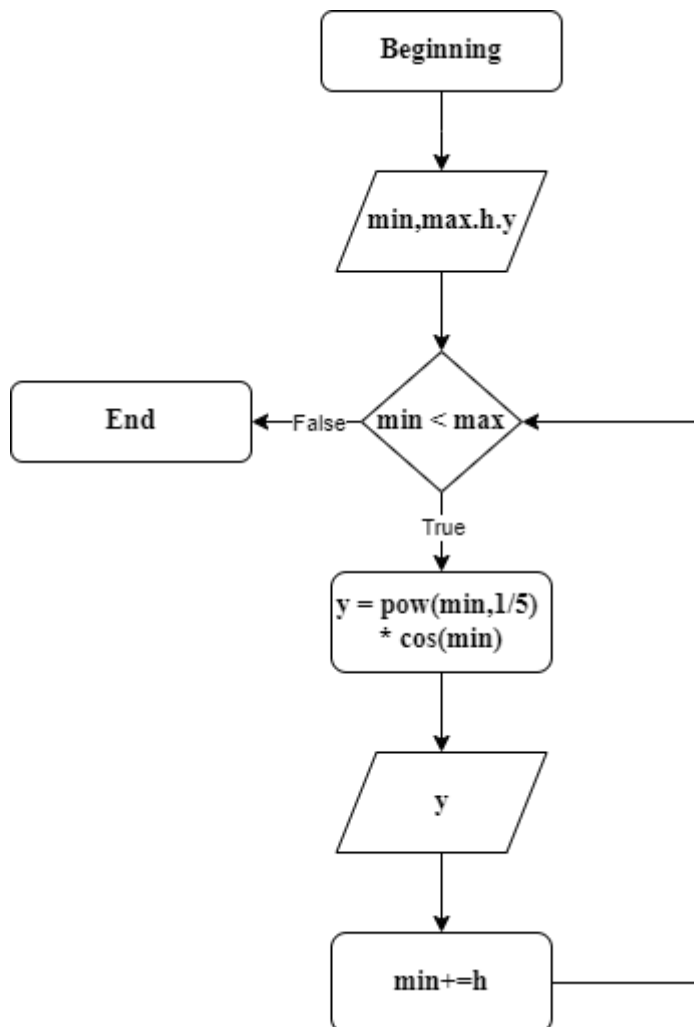
Листинг програми (while):

```
#include <stdio.h>
#include <math.h>
int main() {
    double h = 0.1, y, min = 1, max = 1.5;;
    while (min <= max){
        y = pow(min, 1 / 5) * cos(min);
        printf("%.2lf\n", y);
        min += h;}
    return 0;}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>while.exe
0.54
0.45
0.36
0.27
0.17
```

Блок-схема:



Завдання на самостійну роботу:

5, 10, 15	Скласти програму, що друкує таблицю множення й додавання натуральних чисел у десятковій системі числення.
-----------	---

Таблиця додавання:

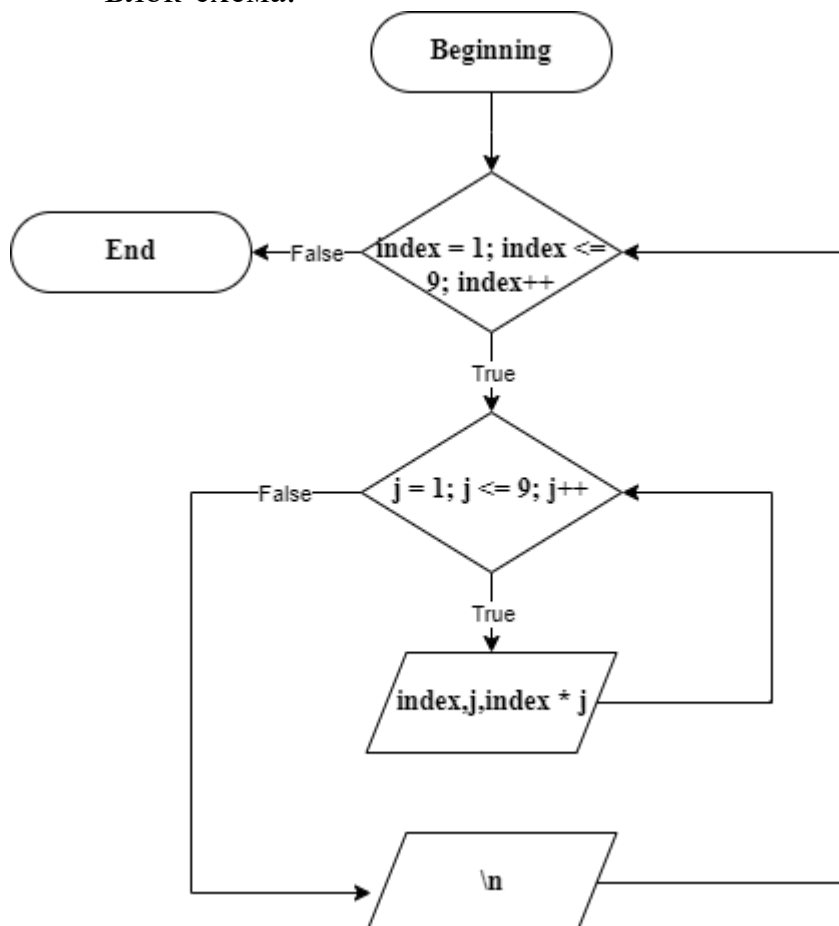
Листинг програми (for):

```
#include <stdio.h>
int main(){
    for(int index = 1; index <= 9; index++){
        for(int j = 1; j <= 9; j++){
            printf("%d * %d = %d\t", index, j, index * j);
        }
        printf("\n");
    }
    return 0;
}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>multiplicationtable.exe
1 * 1 = 1    1 * 2 = 2    1 * 3 = 3    1 * 4 = 4    1 * 5 = 5    1 * 6 = 6    1 * 7 = 7    1 * 8 = 8    1 * 9 = 9
2 * 1 = 2    2 * 2 = 4    2 * 3 = 6    2 * 4 = 8    2 * 5 = 10   2 * 6 = 12   2 * 7 = 14   2 * 8 = 16   2 * 9 = 18
3 * 1 = 3    3 * 2 = 6    3 * 3 = 9    3 * 4 = 12   3 * 5 = 15   3 * 6 = 18   3 * 7 = 21   3 * 8 = 24   3 * 9 = 27
4 * 1 = 4    4 * 2 = 8    4 * 3 = 12   4 * 4 = 16   4 * 5 = 20   4 * 6 = 24   4 * 7 = 28   4 * 8 = 32   4 * 9 = 36
5 * 1 = 5    5 * 2 = 10   5 * 3 = 15   5 * 4 = 20   5 * 5 = 25   5 * 6 = 30   5 * 7 = 35   5 * 8 = 40   5 * 9 = 45
6 * 1 = 6    6 * 2 = 12   6 * 3 = 18   6 * 4 = 24   6 * 5 = 30   6 * 6 = 36   6 * 7 = 42   6 * 8 = 48   6 * 9 = 54
7 * 1 = 7    7 * 2 = 14   7 * 3 = 21   7 * 4 = 28   7 * 5 = 35   7 * 6 = 42   7 * 7 = 49   7 * 8 = 56   7 * 9 = 63
8 * 1 = 8    8 * 2 = 16   8 * 3 = 24   8 * 4 = 32   8 * 5 = 40   8 * 6 = 48   8 * 7 = 56   8 * 8 = 64   8 * 9 = 72
9 * 1 = 9    9 * 2 = 18   9 * 3 = 27   9 * 4 = 36   9 * 5 = 45   9 * 6 = 54   9 * 7 = 63   9 * 8 = 72   9 * 9 = 81
```

Блок-схема:



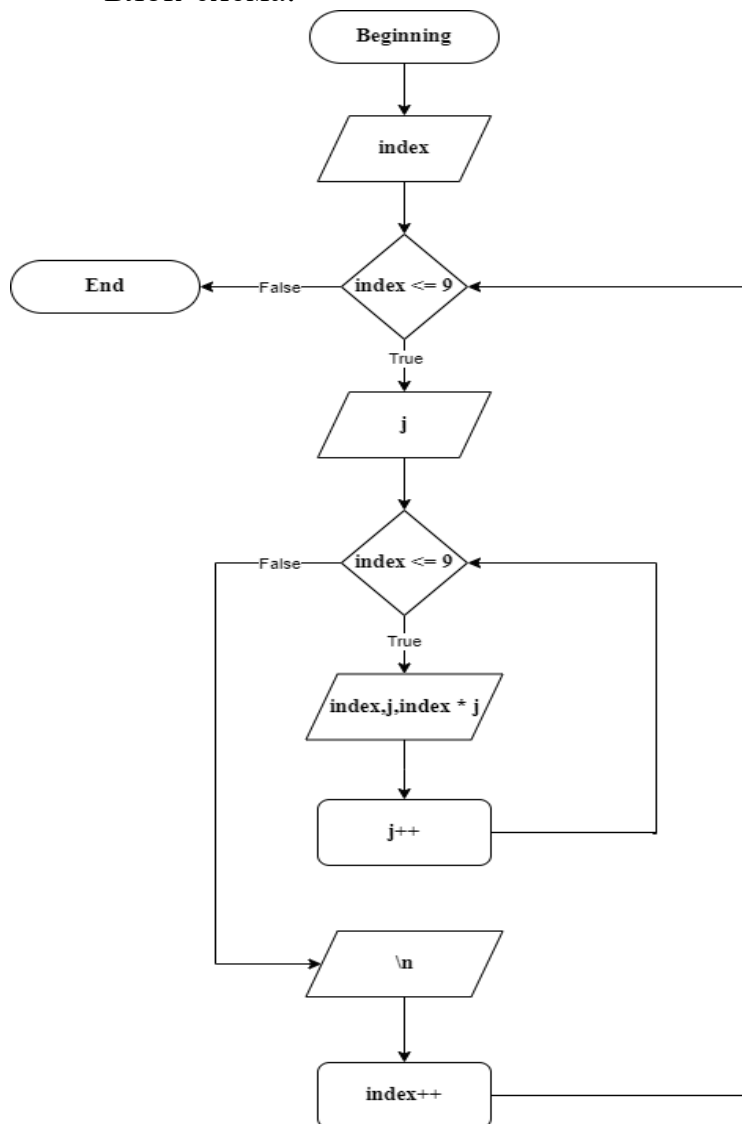
Листинг програми (while):

```
#include <stdio.h>
int main(){
    int index = 1;
    while(index <= 9){
        int j = 1;
        while(j <= 9){
            printf("%d + %d = %d\t",index,j,index * j);
            j++;}
        printf("\n");
        index++;}
    return 0;}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>multiplicationtable.exe
1 * 1 = 1    1 * 2 = 2    1 * 3 = 3    1 * 4 = 4    1 * 5 = 5    1 * 6 = 6    1 * 7 = 7    1 * 8 = 8    1 * 9 = 9
2 * 1 = 2    2 * 2 = 4    2 * 3 = 6    2 * 4 = 8    2 * 5 = 10   2 * 6 = 12   2 * 7 = 14   2 * 8 = 16   2 * 9 = 18
3 * 1 = 3    3 * 2 = 6    3 * 3 = 9    3 * 4 = 12   3 * 5 = 15   3 * 6 = 18   3 * 7 = 21   3 * 8 = 24   3 * 9 = 27
4 * 1 = 4    4 * 2 = 8    4 * 3 = 12   4 * 4 = 16   4 * 5 = 20   4 * 6 = 24   4 * 7 = 28   4 * 8 = 32   4 * 9 = 36
5 * 1 = 5    5 * 2 = 10   5 * 3 = 15   5 * 4 = 20   5 * 5 = 25   5 * 6 = 30   5 * 7 = 35   5 * 8 = 40   5 * 9 = 45
6 * 1 = 6    6 * 2 = 12   6 * 3 = 18   6 * 4 = 24   6 * 5 = 30   6 * 6 = 36   6 * 7 = 42   6 * 8 = 48   6 * 9 = 54
7 * 1 = 7    7 * 2 = 14   7 * 3 = 21   7 * 4 = 28   7 * 5 = 35   7 * 6 = 42   7 * 7 = 49   7 * 8 = 56   7 * 9 = 63
8 * 1 = 8    8 * 2 = 16   8 * 3 = 24   8 * 4 = 32   8 * 5 = 40   8 * 6 = 48   8 * 7 = 56   8 * 8 = 64   8 * 9 = 72
9 * 1 = 9    9 * 2 = 18   9 * 3 = 27   9 * 4 = 36   9 * 5 = 45   9 * 6 = 54   9 * 7 = 63   9 * 8 = 72   9 * 9 = 81
```

Блок-схема:



		Семенчук О. А.		
		Чижмоторя О. В.		
Змн.	Арк.	№ докум.	Підпис	Дата

ДУ «Житомирська політехніка».23.121.10.000 – Лр8

Арк.

9

Таблиця множення:

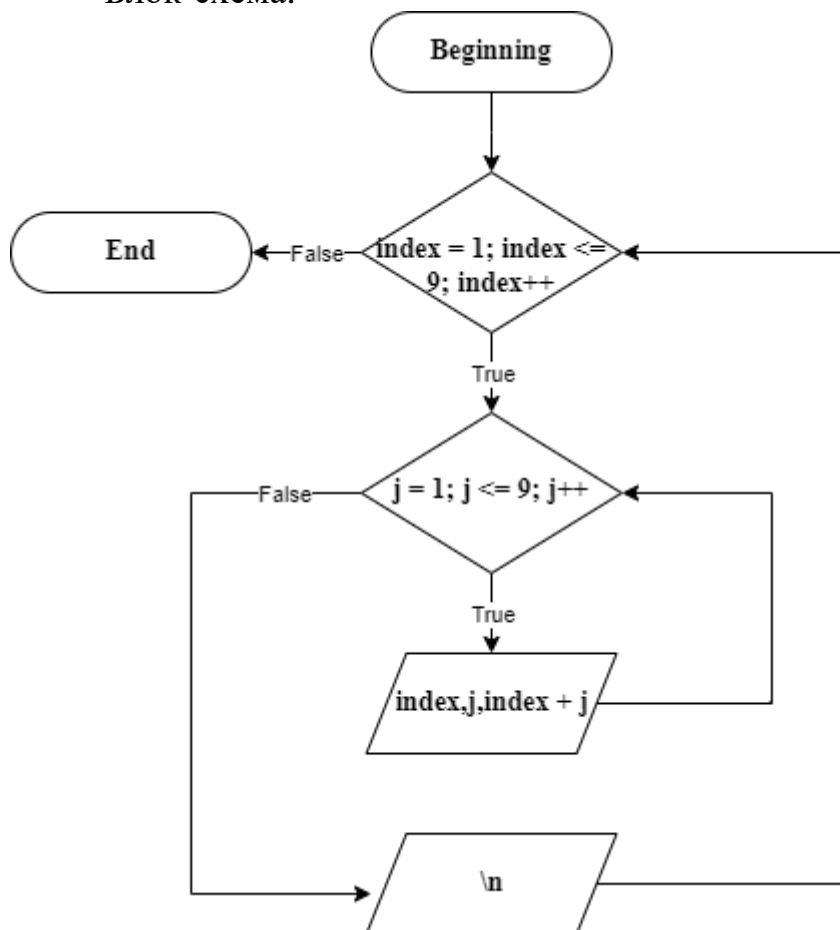
Листинг програми (for):

```
#include <stdio.h>
int main(){
    for(int index = 1; index <= 9; index++){
        for(int j = 1; j <= 9; j++){
            printf("%d + %d = %d\t", index, j, index + j);
        }
        printf("\n");
    }
    return 0;
}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>additiontable.exe
1 + 1 = 2    1 + 2 = 3    1 + 3 = 4    1 + 4 = 5    1 + 5 = 6    1 + 6 = 7    1 + 7 = 8    1 + 8 = 9    1 + 9 = 10
2 + 1 = 3    2 + 2 = 4    2 + 3 = 5    2 + 4 = 6    2 + 5 = 7    2 + 6 = 8    2 + 7 = 9    2 + 8 = 10   2 + 9 = 11
3 + 1 = 4    3 + 2 = 5    3 + 3 = 6    3 + 4 = 7    3 + 5 = 8    3 + 6 = 9    3 + 7 = 10   3 + 8 = 11   3 + 9 = 12
4 + 1 = 5    4 + 2 = 6    4 + 3 = 7    4 + 4 = 8    4 + 5 = 9    4 + 6 = 10   4 + 7 = 11   4 + 8 = 12   4 + 9 = 13
5 + 1 = 6    5 + 2 = 7    5 + 3 = 8    5 + 4 = 9    5 + 5 = 10   5 + 6 = 11   5 + 7 = 12   5 + 8 = 13   5 + 9 = 14
6 + 1 = 7    6 + 2 = 8    6 + 3 = 9    6 + 4 = 10   6 + 5 = 11   6 + 6 = 12   6 + 7 = 13   6 + 8 = 14   6 + 9 = 15
7 + 1 = 8    7 + 2 = 9    7 + 3 = 10   7 + 4 = 11   7 + 5 = 12   7 + 6 = 13   7 + 7 = 14   7 + 8 = 15   7 + 9 = 16
8 + 1 = 9    8 + 2 = 10   8 + 3 = 11   8 + 4 = 12   8 + 5 = 13   8 + 6 = 14   8 + 7 = 15   8 + 8 = 16   8 + 9 = 17
9 + 1 = 10   9 + 2 = 11   9 + 3 = 12   9 + 4 = 13   9 + 5 = 14   9 + 6 = 15   9 + 7 = 16   9 + 8 = 17   9 + 9 = 18
```

Блок-схема:



Листинг програми (while):

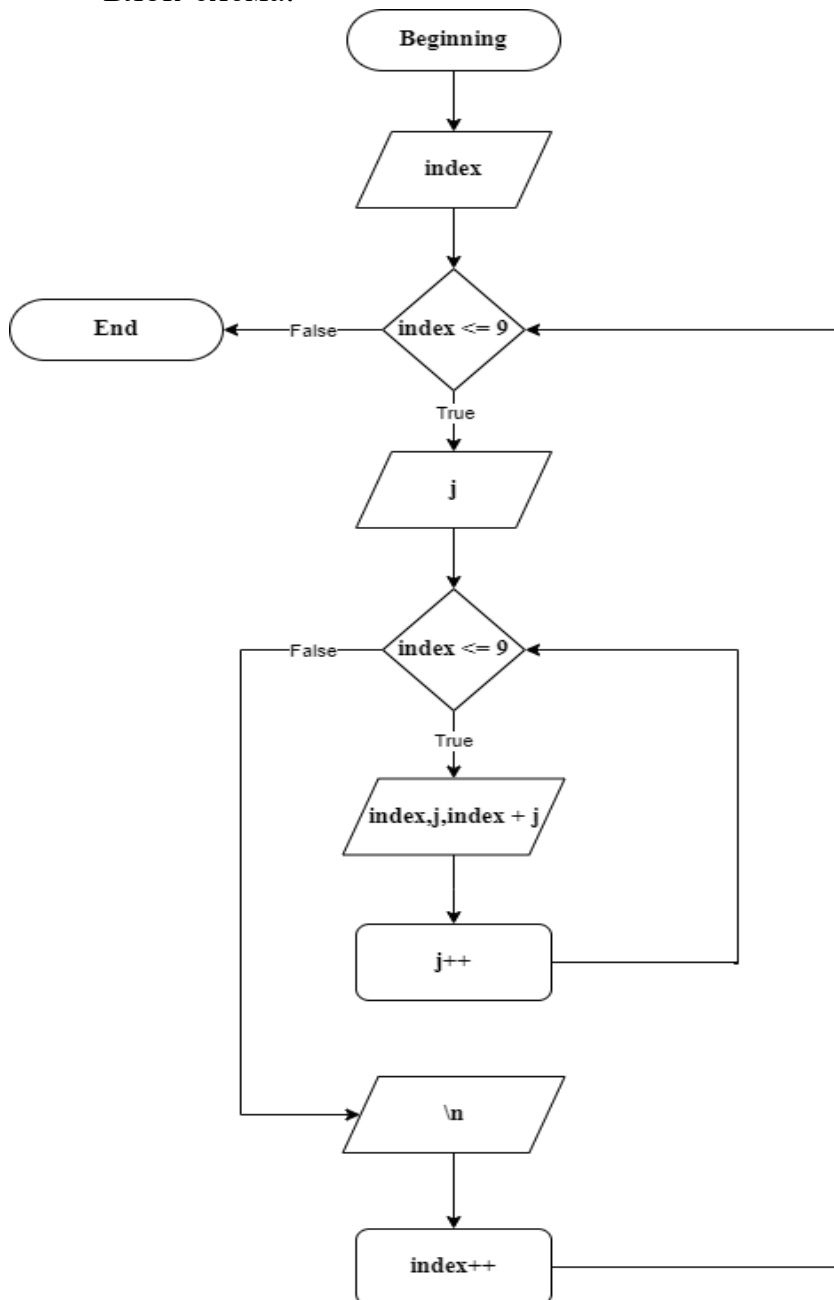
```
#include <stdio.h>
int main(){
    int index = 1;
    while(index <= 9){
        int j = 1;
        while(j <= 9){
            printf("%d + %d = %d\t", index, j, index + j);
            j++;
        }
    }
}
```

```
printf("\n");
index++;}
return 0;}
```

Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>additiontable.exe
1 + 1 = 2    1 + 2 = 3    1 + 3 = 4    1 + 4 = 5    1 + 5 = 6    1 + 6 = 7    1 + 7 = 8    1 + 8 = 9    1 + 9 = 10
2 + 1 = 3    2 + 2 = 4    2 + 3 = 5    2 + 4 = 6    2 + 5 = 7    2 + 6 = 8    2 + 7 = 9    2 + 8 = 10   2 + 9 = 11
3 + 1 = 4    3 + 2 = 5    3 + 3 = 6    3 + 4 = 7    3 + 5 = 8    3 + 6 = 9    3 + 7 = 10   3 + 8 = 11   3 + 9 = 12
4 + 1 = 5    4 + 2 = 6    4 + 3 = 7    4 + 4 = 8    4 + 5 = 9    4 + 6 = 10   4 + 7 = 11   4 + 8 = 12   4 + 9 = 13
5 + 1 = 6    5 + 2 = 7    5 + 3 = 8    5 + 4 = 9    5 + 5 = 10   5 + 6 = 11   5 + 7 = 12   5 + 8 = 13   5 + 9 = 14
6 + 1 = 7    6 + 2 = 8    6 + 3 = 9    6 + 4 = 10   6 + 5 = 11   6 + 6 = 12   6 + 7 = 13   6 + 8 = 14   6 + 9 = 15
7 + 1 = 8    7 + 2 = 9    7 + 3 = 10   7 + 4 = 11   7 + 5 = 12   7 + 6 = 13   7 + 7 = 14   7 + 8 = 15   7 + 9 = 16
8 + 1 = 9    8 + 2 = 10   8 + 3 = 11   8 + 4 = 12   8 + 5 = 13   8 + 6 = 14   8 + 7 = 15   8 + 8 = 16   8 + 9 = 17
9 + 1 = 10   9 + 2 = 11   9 + 3 = 12   9 + 4 = 13   9 + 5 = 14   9 + 6 = 15   9 + 7 = 16   9 + 8 = 17   9 + 9 = 18
```

Блок-схема:



Висновок : У ході лабораторної роботи було успішно вивчено та практично застосовано оператори циклу for та while для різних обчислень та створення таблиць. Це покращило розуміння принципів циклічних алгоритмів та розвинуло навички програмування мовою C.