

Лабораторна робота №4. (2 год.)

Тема: Використання циклів у *Visual Basic*. Складання програм, що містять цикли.

Мета: Розглянути структури циклів у *Visual Basic*. Навчитись складати циклічні програми.

Теоретичні відомості

Цикли (англ. *Loop*¹) – це програмні структури, що забезпечують виконання одного чи декількох операторів задану кількість разів. Наприклад, для кожного числа від 1 до 4 необхідно виконати наступні дії: додати 15; помножити на 3 і піднести до 4-го степеня. Для того, щоб не вводити кожний раз число, для якого визначена ця послідовність операцій використовуються структура циклу, яка може бути записана так:

Для кожного X від 1 до 4 виконати:

$$\begin{aligned} Y &= X + 15 \\ Z &= X \times 3 \\ V &= X^4 \end{aligned}$$

Взяти нове значення X

Процес виконання всіх операторів, що входять в структуру циклу, один раз називається ітерацією (*iteration*). У нашому прикладі здійснюється 4 ітерації. На кожній ітерації виконуватимуться дії:

1) $X=1$ $Y=1+15=16$ $Z=1 \times 3=3$ $V=1^4=1$ $X=2$	2) $X=2$ $Y=2+15=17$ $Z=2 \times 3=6$ $V=2^4=16$ $X=3$	3) $X=3$ $Y=3+15=18$ $Z=3 \times 3=9$ $V=3^4=81$ $X=4$	4) $X=4$ $Y=4+15=19$ $Z=4 \times 3=12$ $V=4^4=256$ $X=5$ (вихід з циклу)
---	--	--	--

Якщо кількість ітерацій визначена, то цикли називаються циклами з фіксованим числом ітерацій (*fixed iteration*). У випадках, коли кількість ітерацій не обмежується (не виконується умова виходу із циклу), цикл називається невизначеним (*indefinite loop*). Блок операторів всередині циклу називається тілом (*body*) циклу.

Найпростішою структурою циклу є фіксовані цикли: *For...Next* і *For Each...Next*. Їх синтаксис має наступний вид:

¹ у лабораторній подано англійський переклад деяких термінів з метою кращого розуміння повідомлень, що формуються під час налагоджування програм

```
For (лічильник) = (початкове значення) to (кінцеве значення)
[Step (крок)]
    (Вираз1)
    (Вираз2)
    ...
    (Вираз ...)
```

Next (лічильник)

Якщо не задавати значення для кроку, то у наступній ітерації значення лічильнику буде змінюватися на 1.

Для прикладу, поданого вище у *Visual Basic* цикл запишеться так:

```
For X=1 to 4
    Y=X+15
    Z=X*3
    V=X^4
Next X
```

Для того, щоб значення лічильника зменшувалося необхідно задати від'ємне значення для кроку.

Цикл *For Each...Next* виконується стільки разів скільки об'єктів є у деякому наборі даних (наприклад, для елементів деякого масиву). Синтаксис циклу має вид:

```
For Each (елемент) In (група)
    (Вираз1)
    (Вираз2)
    ...
    (Вираз ...)
```

Next (елемент)

Також у *Visual Basic* часто використовується цикл *Do...Loop*, який має декілька різновидів:

1) цикл виконується, поки виконується умова. Синтаксис:

```
Do While (умова виконується)
    (Вираз1)
    (Вираз2)
    ...
    (Вираз ...)
```

Loop

Для нашого прикладу структура такого циклу матиме вид:

```
X=1
Do While X <= 4
    Y=X+15
    Z=X*3
    V=X^4
    X=X+1
Loop
```

2) істинність (правдивість) умови перевіряється в кінці циклу. В такій конструкції оператори циклу виконуються як мінімум один раз. Синтаксис:

```
Do
  (Вираз1)
  (Вираз2)
  ...
  (Вираз ... )
Loop While (умова виконується)
```

Для нашого прикладу цикл матиме вид:

```
X=1
Do
  Y=X+15
  Z=X*3
  V=X^4
  X=X+1
Loop While X<4
```

Дві інші структури аналогічні до тих, що ми розглянули за тією відмінністю, що вони перевіряють не правдивість умови, а фальшивість (невиконання умови).

3) Синтаксис:

```
Do Until (умова не виконується)
  (Вираз1)
  (Вираз2)
  ...
  (Вираз ... )
Loop
```

Для нашого прикладу структура такого циклу матиме вид:

```
X=1
Do Until X>4
  Y=X+15
  Z=X*3
  V=X^4
  X=X+1
Loop
```

4) Синтаксис:

```
Do
  (Вираз1)
  ...
  (Вираз N ... )
Loop Until (умова не виконується)
```

Для нашого прикладу цикл матиме вид:

```
X=1
Do
    Y=X+15
    Z=X*3
    V=X^4
    X=X+1
Loop Until X>=4
```

Для виходу із циклу у будь-якому місці використовується команда *Exit*.

Розглянемо використання циклу *For...Next* для вирішення прикладу:

$$y = \sum_{x=1}^2 \frac{\ln(x+1)^{x-1}}{(x+3)^{x+1}}, \text{ step} := 0.1$$

Розмістіть на формі об’єкти та змініть їх властивості так, щоб утворилась форма виду (рис.1):

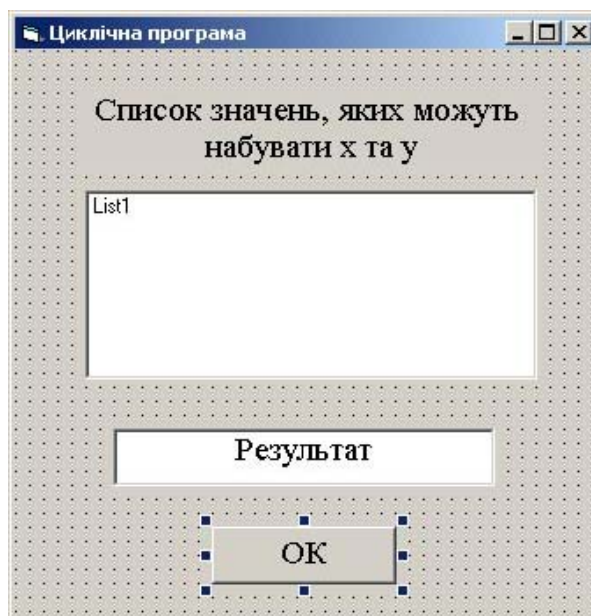


Рис.1

За умови подвійного натиснення мишкою кнопки „OK” переходимо у вікно програми (рис.2):

```

Private Sub Command1_Click()
  Dim y, x As Single
  y = 0

  For x = 1 To 2.1 Step 0.1
    y = y + (Log(x + 1) ^ (x - 1)) / ((x + 3) ^ (x + 1))
    List1.AddItem ("x = " + Str(x) + " y = " + Str(y))
  Next x

End Sub

```

Рис.2

Результат роботи програми показано на рис. 3.

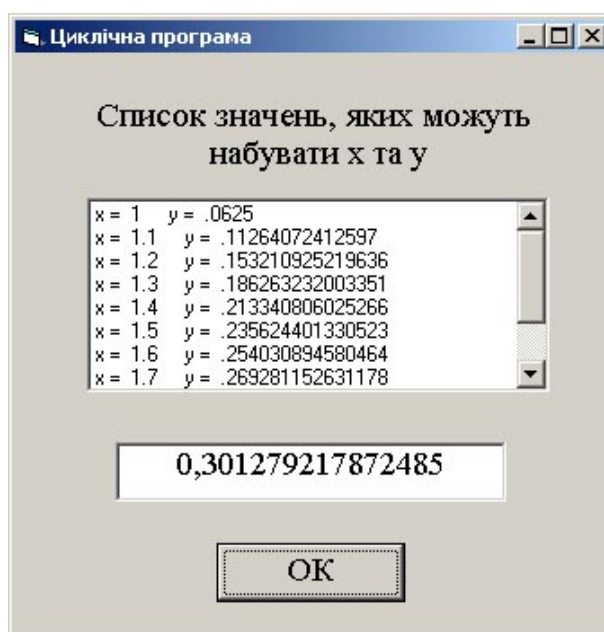


Рис.3

Порядок виконання роботи

1. Відтворити проект, поданий у прикладі.
2. Скласти циклічну програму, для парних номерів варіантів використовуючи цикл *For...Next*, для непарних значень варіантів – цикл *Do...Loop*:

Варіант	Вираз
1	$y = \sum_{x=-2}^2 \frac{x^2(1+x^3)}{10-x^3}, \text{ step} := 0.1$
2	$y = \prod_{x=0.2}^4 \frac{2}{1+e^x}, \text{ step} := 0.2$
3	$y = \sum_{x=0}^{2\pi} \frac{1}{\sqrt[4]{1+\sin^4 x}}, \text{ step} := \frac{\pi}{10}$

4	$y = \prod_{x=0.2}^4 \frac{e^{\frac{x}{2}}}{1+x^2}, \text{ step} := 0.1$
5	$y = \sum_{x=0.1}^{1.5} \frac{\text{tg}^4 x}{x}, \text{ step} := 0.2$
6	$y = \prod_{x=0.2}^4 \frac{1}{e^{x \ln x}}, \text{ step} := 0.1$
7	$y = \sum_{x=0}^{2\pi} \frac{\sin^3 x}{\cos^2 x}, \text{ step} := \pi/10$
8	$\prod_{x=0.2}^4 x^2 \ln^3 x, \text{ step} := 0.1$
9	$y = \sum_{x=0.1}^4 \frac{1}{(x^3 + 1) \cdot \sqrt[5]{(1-x)^2}}, \text{ step} := 0.2$
10	$\prod_{x=1.1}^4 \frac{\sqrt{x}}{\ln^2 x}, \text{ step} := 0.1$
11	$y = \sum_{x=0}^{\pi} \frac{x^x \cdot \sin x^3}{\cos^2 x}, \text{ step} := \pi/2$
12	$\prod_{x=0.2}^4 x^5 \cdot (\sin^2 x - \ln^3 x), \text{ step} := 0.2$
13	$y = \sum_{x=0}^{2\pi} \frac{\sin 3x^4}{\sqrt{1 + \cos^4 x}}, \text{ step} := \pi/10$
14	$\prod_{x=1.1}^4 \frac{\sqrt{x^5 - \ln^2 x}}{\ln^3 x}, \text{ step} := 0.3$
15	$y = \sum_{x=0}^{2\pi} \frac{\text{tg}^3 x}{\cos^2 x + \sin^3 x}, \text{ step} := \pi/10$

3. Провести налагоджування програми. Записати у звіт всі значення, яких можуть набувати x та y в процесі виконання програми.

Звіт з лабораторної повинен містити:

- титульний лист;
- мету;
- блок-схему алгоритму;
- схематичний вид розташування об’єктів на формі;
- програму;
- висновки.

Контрольні запитання

1. Що таке цикли?
2. Що таке ітерація?
3. Які існують види циклів?
4. Наведіть приклади використання циклів для розв'язку економічних задач (задач бухгалтерського обліку, економічного аналізу, тощо).
5. Який синтаксис циклів *For...Next* і *For Each...Next*?
6. Який синтаксис циклів *Do...Loop*?
7. Яка послідовність складання циклічних програм?
8. Як провести налагоджування циклічної програми?