Лабораторна робота № 4. Розгалуження програм. Використання умовних операторів.

Використання **умовного оператора** є одним із самих популярних методів, що змінюють природній порядок виконання операторів програми. Ця алгоритмічна структура представляє розгалуження алгоритму залежно від значення (істинності або хибності) деякої умови. У загальному виді синтаксична конструкція виглядає в такий спосіб:

Якщо *умова* **то** *оператор1* **інакше** *оператор2*,

і читається як «якщо умова дійсна, то виконати оператор1, інакше (якщо умова хибна), виконати оператор2». В VBA розрізняють **рядковий** та **блоковий** умовні оператори**.** *Рядковий* умовний оператор використовується в тому випадку, коли при розгалуженні програми необхідно на кожній гілці виконати по одному операторові. Синтаксис рядкового оператора виглядає наступним чином

```
If умова Then [оператор1] [Else оператор2].
```

Блоковий умовний оператор необхідний тоді, коли операторів декілька й синтаксис оператора буде наступний:

```
If умова Then
  [Блок операторів]
[Else
  [Блок операторів]]
End If
```

Обов'язковий параметр *умова* — являє собою логічне вираження, яке повертає значення TRUE (істина) або FALSE (неправда). Логічний вираз може бути простим або складним. Складні умови утворюються за допомогою логічних операцій *and*, *or*, *not*. Якщо *умова* рівна TRUE, то виконується *оператор1*, якщо *умова* рівна FALSE, те виконується *оператор2*. Далі виконується інструкція, наступна по порядку за рядком з інструкцією If... Then...Else.

Частина оператора If, а саме ключове слово Else і наступні за ним інструкції можна вилучити, тоді за умови рівному FALSE виконується оператор, наступний по порядку за рядком з інструкцією If. Тобто синтаксис оператора виглядає так:

```
If умова Then [оператор]
```

На блок-схемах ці два випадки зображуються так як наведено на рис 5.1.

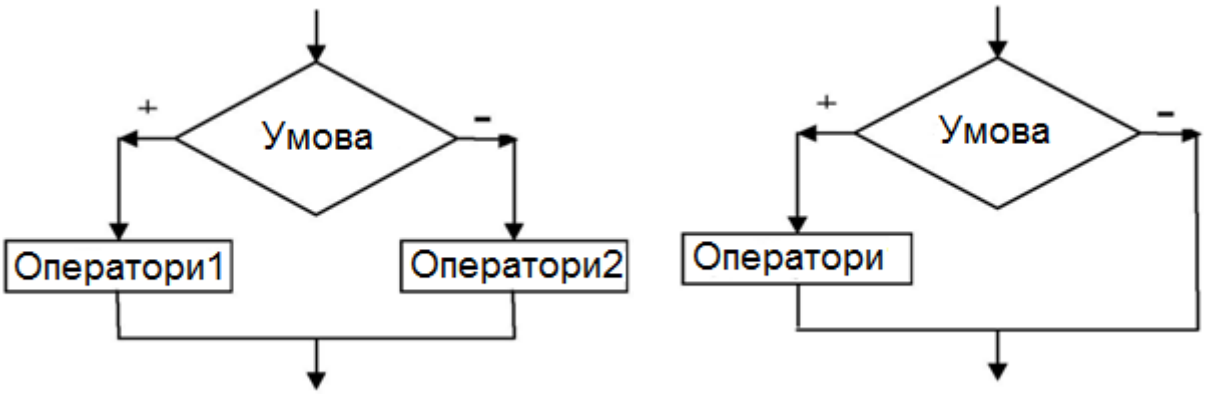


Рис. 5.1. Блок-схеми умовного оператора If... Then...Else

Ще одним умовним оператором в VBA є оператор Select Case. Це оператор вибору з багатьох можливих варіантів. Оператор Select Case виконує одну з декількох груп операторів залежно від значення вираження. У мовах високого рівня такий оператор називається перемикачем. Блок-схема алгоритму наведена на рис 5.2, а синтаксис інструкції Select Case наведен нижче:

```
Select Case вираз
  [Case список умов-1
     [оператори-1]]
 [Case список умов-2
     [оператори-2]]
...
 [Case список умов-n
      [оператори-n]]
 [Case Else
      [оператори_ else]]
End Select
```

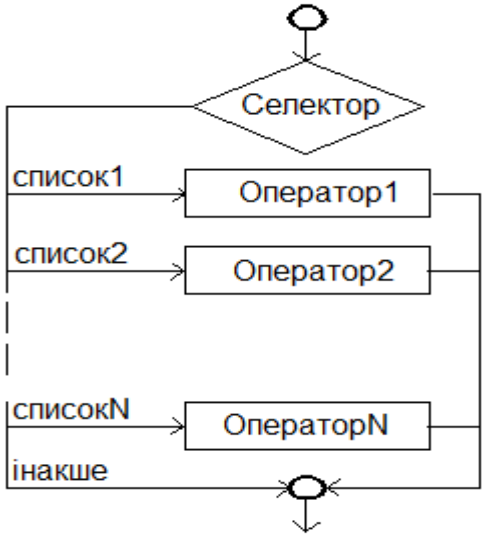


Рис. 5.2. Блок-схеми умовного оператора Select Case

Обов'язковий параметр *вираз* може бути будь-яким виразом будь-якого типу даних VBA, включаючи числовий, строковий, або логічний. *Список умов* — обов'язковий при наявності інструкції CASE. Містить результати обчислення вираження, записані в одному з видів:

- **Case** *КОНСТАНТА1, КОНСТАНТА2, КОНСТАНТА3, ...*
- **Case Is** *ЗНАК_ВІДНОСИНИ КОНСТАНТА*
- **Case** *КОНСТАНТА1* **to** *КОНСТАНТА2*

Виконується оператор Select Case у такий спосіб — спочатку обчислюється значення виразу, який знаходиться після ключових слів Select Case, потім проводиться перевірка на збіг значення виразу й одного з Case-умов, у випадку

збігу виконуються оператори, поміщені за цим Case-умовою. У випадку, якщо збігу не виявлено у жодному з Case-умов, то виконуються оператори, що перебувають після ключових слів Case Else. Якщо список виражень у Case-умові записаний у першій формі, то необхідно, щоб результат обчислень вираження рівнявся однієї константі зі списку; якщо Case-умова записана в другій формі, то необхідно, щоб виконувалася умова відносини між значенням вираження й константою; якщо у третій формі, то необхідне виконання співвідношення: *КОНСТАНТА1<=вираження<=КОНСТАНТА2*.

Завдання 1

Визначити радіус кругової пожежі, якщо відома лінійна швидкість поширення вогню й час горіння. Необхідно врахувати: якщо час горіння перевищує 10 хв., то швидкість поширення вогню збільшується на 50%. Якщо об'єкт відноситься до пожеженебезпечної зони, то радіус пожежі збільшується на 20%.

Аналіз проекту

Для реалізації проекту введемо наступні позначення. Тип *Single*: *V* – швидкість поширення пожежі, *T* – час горіння; *R* – радіус пожежі. Тип *Boolean*: *Fire* – наявність того, що об'єкт відноситься до пожеженебезпечної зони.

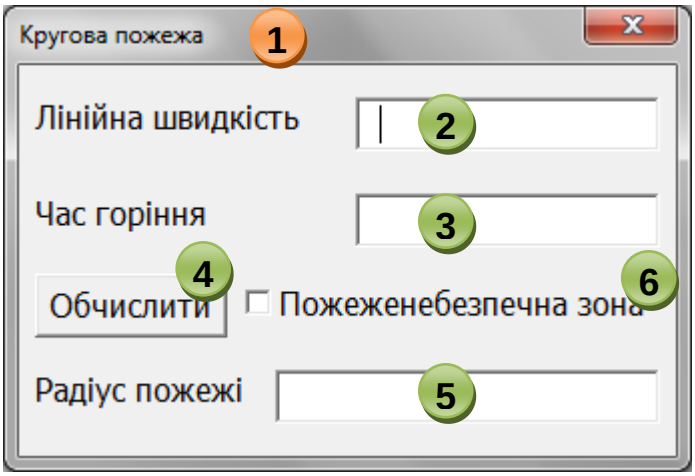
Спочатку визначаємо швидкість горіння. Якщо *T>10*, то $V_г=1,5\ V$, інакше $V_г=V$. Радіус пожежі визначається як $R=TV_г$. Якщо об'єкт відноситься до пожеженебезпечної зони (*Fire=True*), то $R_п=1,2\ R$, *інакше* $R_п=R$.

Порядок виконання

1. Перебуваючи в середовищі *Ms Word* або іншого додатку з *Microsoft Office*, натисніть **ALT+ F11**, щоб перейти в середовище VBA.

2. На панелі інструментів клацніть на кнопку **Insert Userform** . З'являється вікно з формою *Userform1*.

3. Створіть форму згідно з ескізом, установивши у вікні властивостей **Properties** значення властивостей (**Name**) і **Caption**, які наведені нижче. Зверніть увагу на те, що елемент керування прапорець створюється за допомогою кнопки панелі керування, яка має вигляд . На формі він має ім'я *chkfire*, а властивість **Caption** — *Пожеженебезпечна зона.*



Властивість (**Name**)

1 — *frmFire*

2 — *txtV*

3 — *txtT*

4 — *CmdCalculate*

5 — *txtR*

6 — *chkFire*

4. Перейдіть у вікно коду, двічі клацнувши кнопку на **Обчислити**, і введіть наступний текст процедури

```
Private Sub cmdCalculate_Click()
Dim V As Single, T As Single, R As Single
'Уведення вихідних даних
V = Csng(txtv)
T = Csng(txtt)
'визначаємо швидкість поширення
If T > 10 Then
V = 1.5 * V
End If
'визначаємо радіус пожежі
R = V * T
'визначаємо радіус пожежі для пожеженебезпечної зони
If chkfire Then
R = 1.2 * R
End If
'Виводимо значення на форму
txtr = Format(R, "0.00") & " метра (ов)."
End Sub
```

5. Запустіть програму на виконання, клацнувши на вільному місці форми, а потім на кнопці **Run** (або натисніть клавішу **F5**).

6. Перевірте правильність роботи створеного додатку при різних вихідних даних.

7. Збережіть документ у файлі *Лаб4_1.doc* у своїй папці на жорсткому диску.

Завдання 2

Обчислити математичну функцію *y(x,a)*, при заданих значеннях *a* і *x*. Вихідні значення ввести із клавіатури.

$$y(x,a)=\begin{cases} ln(a^2+1), \text{ если } x<a; \\ sin(ax), \text{ если } x\geq a. \end{cases}$$

Порядок виконання

1. Перейдіть у середовище VBA, нажавши клавіші **ALT+F11**
2. У вікні властивостей **Project** клацніть на кнопку **View code** (або виконайте подвійне клацання мишкою на пункті **Thisdocument**). З'являється вікно коду VBA.
3. Перейдіть у вікно коду й уведіть наступний текст процедури

```
Sub Primer2()
Dim a, x, y As Single
' Уводимо вихідні дані із клавіатури
a = Val(Inputbox("Уведіть a"))
x = Val(Inputbox("Уведіть x"))
' обчислюємо значення
If x < a Then y = Log(a ^ 2 + 1) Else y = Sin(a * x)
'Виводимо y
Msgbox ("Значення y=" & Str(y))
End Sub
```

4. Запустіть програму на виконання, клацнувши на кнопці **Run** (або натисніть клавішу **F5**).
5. Збережіть документ у файлі *Лаб4_2.doc* у своїй папці на жорсткому диску.

Завдання 3

Написати програму для визначення відповідності між номером дня тижня і його назвою згідно з формулою Зеллера (0-сб, 1-нд, 2-пн, 3-вт, 4-ср, 5-чт, 6-пт).

Порядок виконання

1. Перейдіть у середовище VBA, нажавши клавіші **ALT+F11**
2. У вікні властивостей **Project** клацніть кнопку **View code** (або виконайте подвійне клацання мишкою на пункті **Thisdocument**). З'являється вікно коду VBA.
3. Перейдіть у вікно коду й уведіть наступний текст процедури:

```
Sub Primer3()
Dim Day  As Integer
 'Уводимо із клавіатури номер дня
Day = Val(Inputbox("Уведіть номер дня тижня"))
' Визначаємо назву дня тижня
```

```
Select Case Day
Case 0
Msgbox ("Це субота")     'День із номером 0
Case 1
Msgbox ("Це неділя")     'День із номером 1
Case 2
Msgbox ("Це понеділок") 'День із номером 2
Case 3
Msgbox ("Це вівторок")   'День із номером 3
Case 4
Msgbox ("Це середа")     'День із номером 4
Case 5
Msgbox ("Це четвер")     'День із номером 6
Case 6
Msgbox ("Це п'ятниця")   'День із номером 7
End Select
End Sub
```

4. Запустіть програму на виконання, клацнувши на кнопці **Run** (або натисніть клавішу **F5**).

5. Перевірити правильність роботи програми шляхом уведення різних номерів дня тижня.

6. Доробити програму, передбачивши захист від неправильного введення дня тижня. Для цього перед рядком коду

```
End Select
```

додати рядки:

```
Case Else
Msgbox ("Такого дня тижня не існує")
```

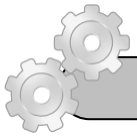
7. Перевірити правильність роботи програми шляхом уведення різних номерів дня тижня, у тому числі й неправильних.

8. Збережіть документ у файлі *Лаб4_3.doc* у своїй папці на жорсткому диску.

Завдання для самостійної роботи

1. Знайти значення функції *y(x)* при різних значеннях *x*, що вводяться із клавіатури.

$$y(x)=\begin{cases} sin(x), & x<-2 \\ 2x, & -2\le x<0 \\ |x|, & 0\le x<2 \\ x^2 & x>2 \end{cases}$$



Результат роботи:

1. Файл *Лаб4_1.doc* з формою *Кругова пожежа*.
2. Файли *Лаб4_2.doc, Лаб4_3.doc* з виконаними завданнями.

Питання для самоконтролю

1. Що таке алгоритми, що розгалужуються?
2. Яким елементом на блок-схемі позначається умовний оператор?
3. Який синтаксис має умовний оператор If... Then...Else і Select Case?
4. У чому відмінність рядкового й блокового операторів If... Then...Else? У яких випадках потрібно використовувати блоковий оператор?
5. У якому виді можна записати список умов в операторі Case?