

Тема 3. Основи структурного програмування

План

1. Теорема структурного програмування, його ознаки
2. Лінійні конструкції структурного програмування: змінні
3. Робота з комірками робочого аркуша Excel

3.1. Теорема структурного програмування, його ознаки

Структурне програмування – це методологія розробки програмного коду, в основі якої покладено його представлення у вигляді ієрархічної структури блоків.

Дана методологія була запропонована у 70-х роках ХХ століття Е. Дейкстрой, пізніше була розроблена та доповнена Н. Віртом.

Дана методологія передбачає:

1. Теорема структурного програмування – будь-який алгоритм можна представити у вигляді лінійних, розгалужених та циклічних конструкцій з використанням змінних.
 - послідовне (лінійне) виконання операцій в рамках структурного програмування передбачає однократне виконання операцій в тому порядку, в якому вони приведені в тексті програми.
 - розгалужене виконання операцій в рамках структурного програмування передбачає однократне виконання однієї, або більше груп операцій, в залежності від результату перевірки деякої заданої умови;
 - циклічне виконання операцій в рамках структурного програмування передбачає багатократне виконання однієї і тієї самої групи операцій, до тих пір, поки виконується деяка задана умова;

В програмі базові конструкції можуть бути вкладеними одна в одну довільним чином, однак, ніяких інших засобів керування послідовністю виконання операцій не передбачається.

2. Властивість модульності в рамках структурного програмування передбачає оформлення фрагментів програм, які багаторазово використовуються на протязі її виконання, у вигляді підпрограм (процедур) та функцій. В этом случае в тексте основной программы, вместо помещённого в подпрограмму фрагмента, вставляется инструкция вызова подпрограммы. Під поняттям «модуль» розуміють функціонально завершену частину програми.
3. Розробка програми ведеться поетапно, методом «зверху вниз». Спочатку пишеться текст основної програми, в якому, замість кожного логічного фрагменту коду, впроваджується виклик підпрограми, яка буде виконувати цей фрагмент.

Замість справжніх, працюючих процедур, в програму впроваджуються «заглушки», які нічого не виконують. Отримана програма проходить перевірку та відладку. Після того, як розробник впевниться, що процедури запускаються в правильній послідовності (тобто загальна структура головного програмного

коду є правильною), процедури-заглушки послідовно змінюються на реально працюючі, причому, розробка кожної процедури ведеться тим самим методом, що й основної програми. Така послідовність гарантує, що на кожному етапі розробки, програміст одночасно має справу з визначеною кількістю зрозумілих фрагментів, та може бути впевненим, що загальна структура всіх більш високих рівнів програми є правильною.

3.2. Лінійні конструкції структурного програмування: змінні

На мові програмування VBA, коментарі можуть бути вставлені в будь-якому місці процедури й вони будуть повністю ігноруватись інтерпретатором.

Способи розміщення коментарів:

1. За допомогою ключового слова `Rem`.

– на початку рядка:

```
Rem Коментарі розробника
```

– після чергової команди, ключове слово `Rem` повинно бути відокремлено від нього двокрапкою:

```
Dim intCounter :Rem оголошення змінної
```

2. За допомогою апострофа, який може бути розташований як на початку рядка, так й в одному рядку з оператором:

```
Dim intCounter ' оголошення змінної
```

Типи даних

Багато мов програмування, у тому числі й Visual Basic, підтримують різні типи даних: цілі, строкові, булеві, з плаваючою крапкою і т.д., див. таблицю.

Підтип	Значення
Boolean	TRUE або FALSE («Істина» або «Брехня»)
Byte	Ціле від 0 до 255
Integer	Ціле -32768 до 32767
Long	Ціле -2147483648 до 2147483647
Currency	Дійсне число з фіксованою крапкою, в діапазоні від: -922337203685477,5808 до 922337203685477,5808
Date(Time)	Число, що представляє дату між 1 січня 100 року і 31 грудня 9999 року
Single	Дійсне число
Double	Дійсне число підвищеної точності
String	Рядок символів змінної довжини; максимальна довжина приблизно 2 мільярди символів

Змінні

Дані зберігаються в змінних. Змінна – це поійменована область пам'яті, вміст якої може бути змінений.

Приклад 1. Використання змінної для виведення повідомлення на екран

```
MessageVar = "Текстове повідомлення на екрані"  
A = MessageVar  
Msgbox (A)
```

Оголошення змінних

Змінні можуть бути оголошені явно, або неявно:

- змінна оголошується неявно самим фактом її використання (див. попередній приклад);
- змінна оголошується явно за допомогою оператора Dim, наприклад:

```
Dim MessageVar [As тип]
```

Додатковий не обов'язковий параметр [As тип] дозволяє оголосити змінну певного типу.

Назви змінних

Назви змінних у VBA повинні відповідати наступним правилам:

- повинні починатися з букви;
- не можуть містити крапки, пробіли й деякі спеціальні символи, такі як: #, ~, /;
- не можуть бути довшими, ніж 255 символів;
- повинні бути унікальними в своїй області видимості (*область видимості обговорюється нижче*);
- зарезервовані слова (if, option, for, dim та ін.) не можуть використовуватись в якості назв змінних.

Назви змінних повинні бути змістовними й описовими по відношенню до їхніх значень. *Тоді програмний код буде бульш читабельним й легше зрозумілим.*

Приклад 2. Використання змінних

```
Year = InputBox ("Скільки вам років?")  
Msgbox ("Вам " & Year & " років")
```

```
A = 15  
B = 6  
A = A + 5  
B = 2 * A  
Msgbox ("A = " & A & "; B = " & B)
```

3.3. Робота з комірками робочого аркуша Excel

Для роботи з комірками робочого аркуша Excel використовується об'єкт Range. Для того, щоб звернутись до певної комірки, слід записати Range (адреса), наприклад:

```
Range ("A5")
```

або

```
Range("Аркуш1!А5")
```

Властивість `Value` використовується для запису, або зчитування значення в комірку (з комірки), наприклад:

```
Range("А5").Value = 10  
Num = Range("А5").Value
```

Крім об'єкта `Range`, до комірок робочого аркуша можна за допомогою об'єкту `Cells`:

```
Worksheets("Аркуш1").Cells(3,4).Value = 5  
Num = Worksheets("Аркуш1").Cells(3,4).Value
```

На відміну від об'єкта `Range`, де адресація комірки складається з назви стовпця й номера рядка, в об'єкті `Cells` вказується номер рядка та номер стовпця. Це зручно при організації обробки масивів даних на робочому аркуші, де посилання на комірку формується в програмі автоматично, у вигляді циклічних конструкцій.

Приклад 3. Робота з комірками

1. Дана колонка з 5 чисел. Записати ці числа в іншій колонці, збільшені у 2 рази;
2. Дана колонка з 5 чисел. Знайти їхню суму й зберегти її на робочому аркуші (за допомогою об'єктів `Range` та `Cells`).
3. Виконати опитування користувача (Ім'я, вік) й зберегти його результати на робочому аркуші;
4. Знайти результат арифметичної операції A^B та зберегти його на робочому аркуші. Вхідні дані також містяться на аркуші.