

УПРАВЛЯЮЩИЕ СТРУКТУРЫ VBA

Управляющие структуры определяют порядок выполнения кода.

В VBA используются следующие структуры управления:

1. [If ... Then ... Else ...;](#)
2. [For ... Next ...;](#)
3. [While ... Wend ...;](#)
4. [Do ... Loop ...;](#)
5. [Select Case ... End Select ...](#)

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ IF ... THEN ... ELSE ...

Эта конструкция изменяет порядок выполнения инструкций в зависимости от результатов проверки заданного условия (или набора условий).

Конструкция:

Строчный If :

If <условие> **Then** <блок операторов 1> **Else**
<блок операторов 2>
End If

УПРАВЛЯЮЩАЯ ИСТРУКЦИЯ ... THEN ... ELSE ...

Конструкция:

Блочный If, если требуется несколько условий:

```
If <условие 1> Then  
  <блок операторов 1>  
Else  
  If <условие 2> Then  
    <блок операторов 3>  
  End If  
End If
```

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ FOR ... NEXT ...

Эта инструкция позволяет выполнять определённые действия нужное количество раз. Такие инструкции, как правило, называют циклами.

Конструкция:

For [счетчик] = начало **To** конец [**Step** шаг]
[инструкции]
Next [счетчик]

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ WHILE ... WEND ...

Инструкция While ... Wend ... выполняет последовательность инструкций, пока заданное условие имеет значение True.

Конструкция:

While <условие>
[инструкции]
Wend

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ DO ... LOOP ...

Эта инструкция позволяет выполнять определённые действия не только до тех пор, пока соблюдается заданное условие, но и наоборот, пока не будет выполнено заданное условие.

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ DO ... LOOP ...

Do While <условие>
[блок Операторов]
Loop

Условие проверяется до того, как выполняется группа операторов, образующих тело цикла. Цикл продолжает свою работу, пока это условие выполняется (то есть имеет значение True), иными словами, в этой конструкции указывается условие продолжения работы цикла

Do
[блок Операторов]
Loop While <условие>

Условие проверяется после того, как операторы, составляющие тело цикла, будут выполнены хотя бы один раз. Цикл продолжает свою работу, пока это условие остается истинным, иными словами, в этой конструкции указывается условие продолжения работы цикла

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ DO ... LOOP ...

<p>Do Until <условие> [блок Операторов] Loop</p>	<p>Условие проверяется до того, как выполняется группа операторов, образующих тело цикла. Цикл продолжает свою работу, если это условие еще не выполнено, и прекращает работу, когда оно станет истинным, иными словами, в этой конструкции указывается условие прекращения работы цикла</p>
<p>Do [блок Операторов] Loop While <условие></p>	<p>Условие проверяется после того, как операторы, составляющие тело цикла, будут выполнены хотя бы один раз. Цикл продолжает свою работу, пока это условие остается истинным, иными словами, в этой конструкции указывается условие продолжения работы цикла</p>

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ SELECT CASE ... END SELECT ...

Эта инструкция в зависимости от результатов значения переменной или выражения выполняет один из нескольких фрагментов кода.

УПРАВЛЯЮЩАЯ ИНСТРУКЦИЯ SELECT CASE ... END SELECT ...

Конструкция:

Select Case [переменные]

Case [первое значение переменной]

<список операторов 1>

Case [второе значение переменной]

<список операторов 2>

Case [N-ое значение переменной]

<список операторов N>

Case Else

<список операторов N+1>

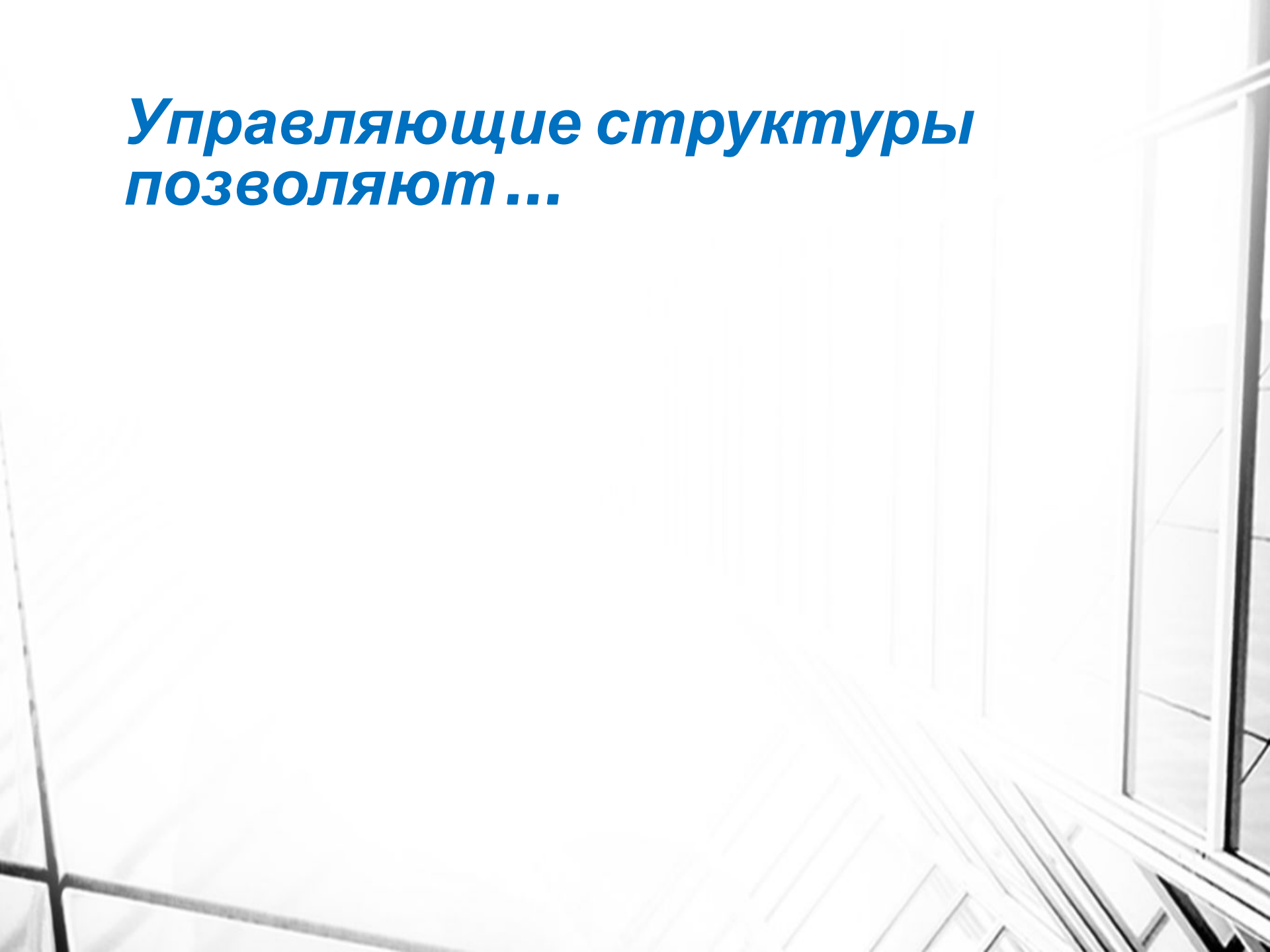
End Select

Структурное программирование — методология разработки программного обеспечения, в основе которой лежит представление программы в виде иерархической структуры блоков.

Управляющие структуры - это основа языков программирования. Они позволяют задать параметр и затем в зависимости от его значения выполнить то или иное действие.

Управляющие структуры определяют порядок выполнения кода и помогают сократить число ошибок, упростить отладку.

***Управляющие структуры
позволяют ...***



Управляющая инструкция IF ... THEN ... ELSE ...

Управляющая инструкция
WHILE ... WEND ...

**Дайте определение
управляющим структурам:
WHILE ... WEND ...
FOR ... NEXT ...**

Visual Basic for Application ***-- Эмо***



Результаты теста :