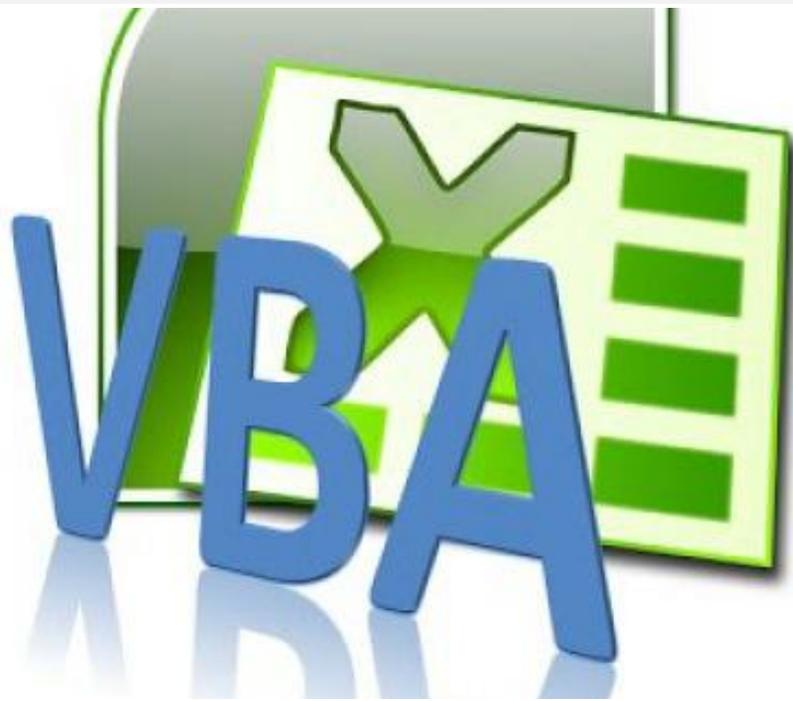
An aerial photograph of a rural agricultural landscape. The scene is dominated by vibrant green terraced fields, likely rice paddies, which are separated by narrow earthen paths. In the center-right of the image, there is a prominent, irregularly shaped pond with a brownish, reflective surface. The surrounding area includes patches of dense tropical forest with various shades of green, and a few palm trees are visible. A white, irregular polygonal shape is overlaid on the image, pointing towards the text on the left.

Реализация алгоритмов в среде программирования.

Часть 2. Программирование на VBA



Программирование на VBA

Создать код на VBA
МОЖНО ДВУМЯ
СПОСОБАМИ:

- ввести вручную с клавиатуры код в редакторе VBA, создав процедуру **Sub** или **Function**;
- использовать средства записи макросов.

Программирование на VBA

Язык VBA является **объектно-ориентированным языком**. Пользователь может создавать свои собственные **классы** (*Classes*), **коллекции** (*Collections*) и **объекты** (*Objects*) с собственными **свойствами** (*:Properties*) и **методами** (*Methods*).

Такие объекты создаются в модулях классов (*class modules*). Рассмотрим структуры управления программой на VBA



Структуры разветвлений

Программа на VBA представляет собой последовательность выполняемых друг за другом инструкций, называемых **операторами**. Одним из базовых операторов является оператор присваивания, который вычисляет некоторое выражение и присваивает результат переменной или объекту. Однако последовательность выполнения операторов можно контролировать: какие-то операторы пропускать, а какие-то повторять повторно, в зависимости от определенных условий. Для этих целей и используются **структуры разветвлений**, в последние годы их также называют структурами принятия решений, или структурами управления потоком (**Flow Control**).



Структуры разветвлений

Идея структурного подхода состоит в том, что программа представляет собой комбинацию или вложение друг в друга стандартных структур, причем каждая структура имеет только один вход и только один выход. Войти в структуру, минуя ее вход и выйти из нее минуя ее выход, нельзя, поскольку это существенно усложнит и работу программы и ее отладку.



Структуры разветвлений

Введем некоторые обозначения. Пусть L и $L1$ – это некоторое отношение (условия), например, $a > b$ или $x > 0$ и т.п. Пусть S , $S1$, $S2$ – некоторые действия, например, $y = x$, или $z = k + p$ и т.д.



Структура развилки

Структурой развилки (структурой выбора) называется такая структура, которая обеспечивает выбор из двух направлений реализуемого процесса. Выбор зависит от условия, имеющегося в данной структуре.

Можно выделить две структуры развилки. Первая из них, назовем ее **структурой двойного ветвления** (или иначе – **стандартное разветвление**), применяется, когда имеются две группы действий и структура обеспечивает выбор одной из них. Второй тип структуры развилки, назовем ее **упрощенной развилкой** (она называется также структурой типа **обход**, или структурой **усеченного разветвления**), используется, когда имеется только одна группа действий и структура либо выбирает их, либо пропускает.



Двойное ветвление

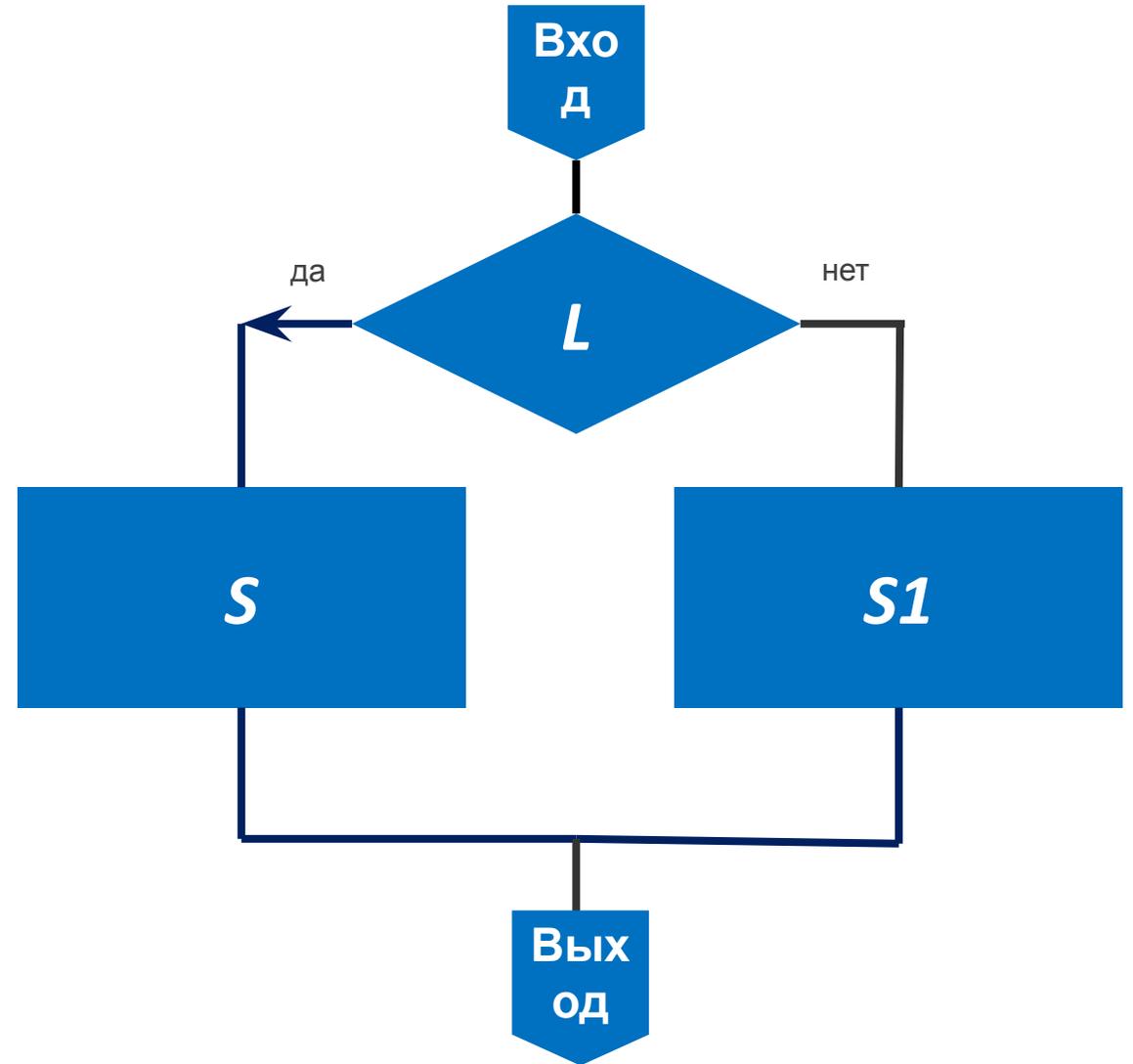
Рассмотрим структуру **двойное ветвление**:

Если условие L истинно, то будет выполнены действия S , а если ложны, то – действия $S1$.

Для программирования структуры может быть использован либо строчный, либо блочный оператор IF. **Строчный оператор** имеет следующих вид:

```
IF L THEN S ELSE S1
```

Этот оператор может быть записан только в одну строку, поэтому если условие L и действие S и $S1$ громоздкие, или S и $S1$ составлены из нескольких операторов, то их просто невозможно разместить в одну строку, необходимо воспользоваться конструкцией **блочный IF**



Двойное ветвление

Полноценную реализацию структуры позволяет осуществить **блочный оператор IF**, который имеет вид:

```
IF L THEN
```

```
  S
```

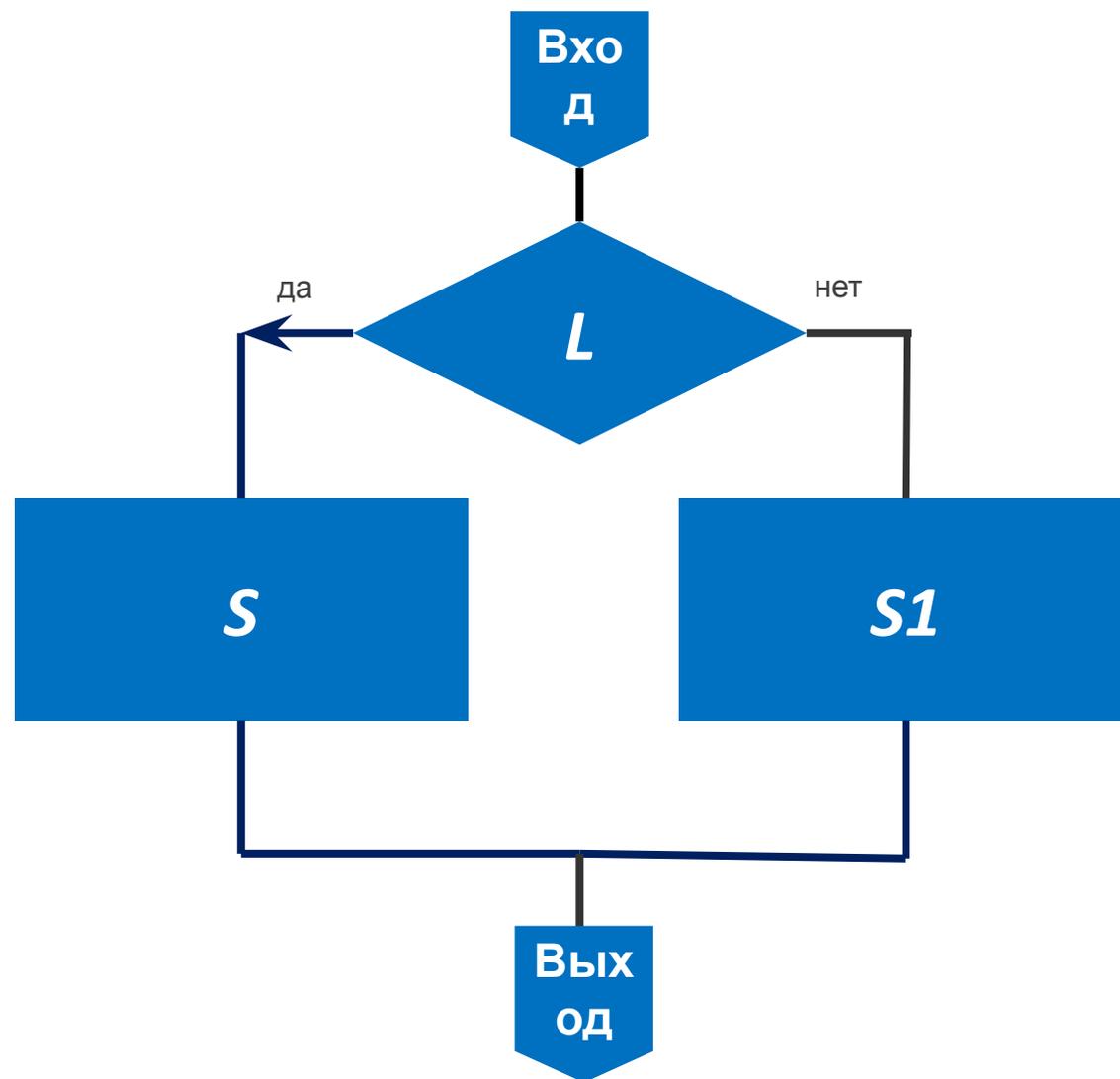
```
ELSE
```

```
  S1
```

```
END IF
```

Зона действия блочного оператора IF хорошо обозрима, поскольку каждый блочный IF всегда заканчивается ключевыми словами END IF.

После ключевых слов THEN, ELSE и END IF в строке больше ничего нельзя писать. В то же время число операторов в группе S и группе S1 ничем не ограничено, что значительно расширяет возможности блочного оператора IF и существенно облегчает процесс программирования, особенно когда алгоритм содержит сложные условия.

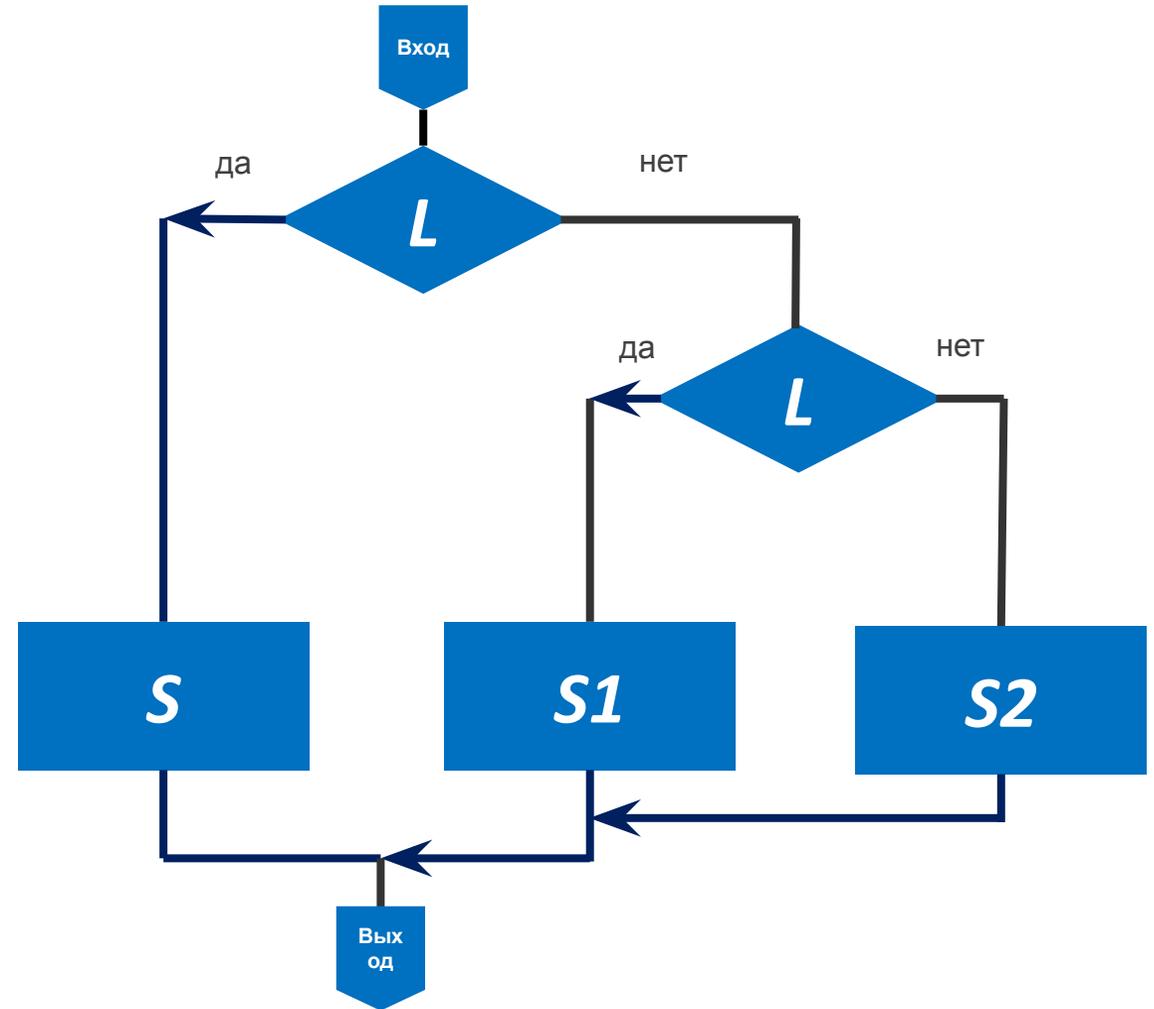


Двойное ветвление

Проиллюстрируем это на примере вложения одной структуры IF в другую такую же структуру IF (пусть вложение необходимо сделать, когда условие L ложно, т.е. при «нет» на блок-схеме).

Программа для этого вложения:

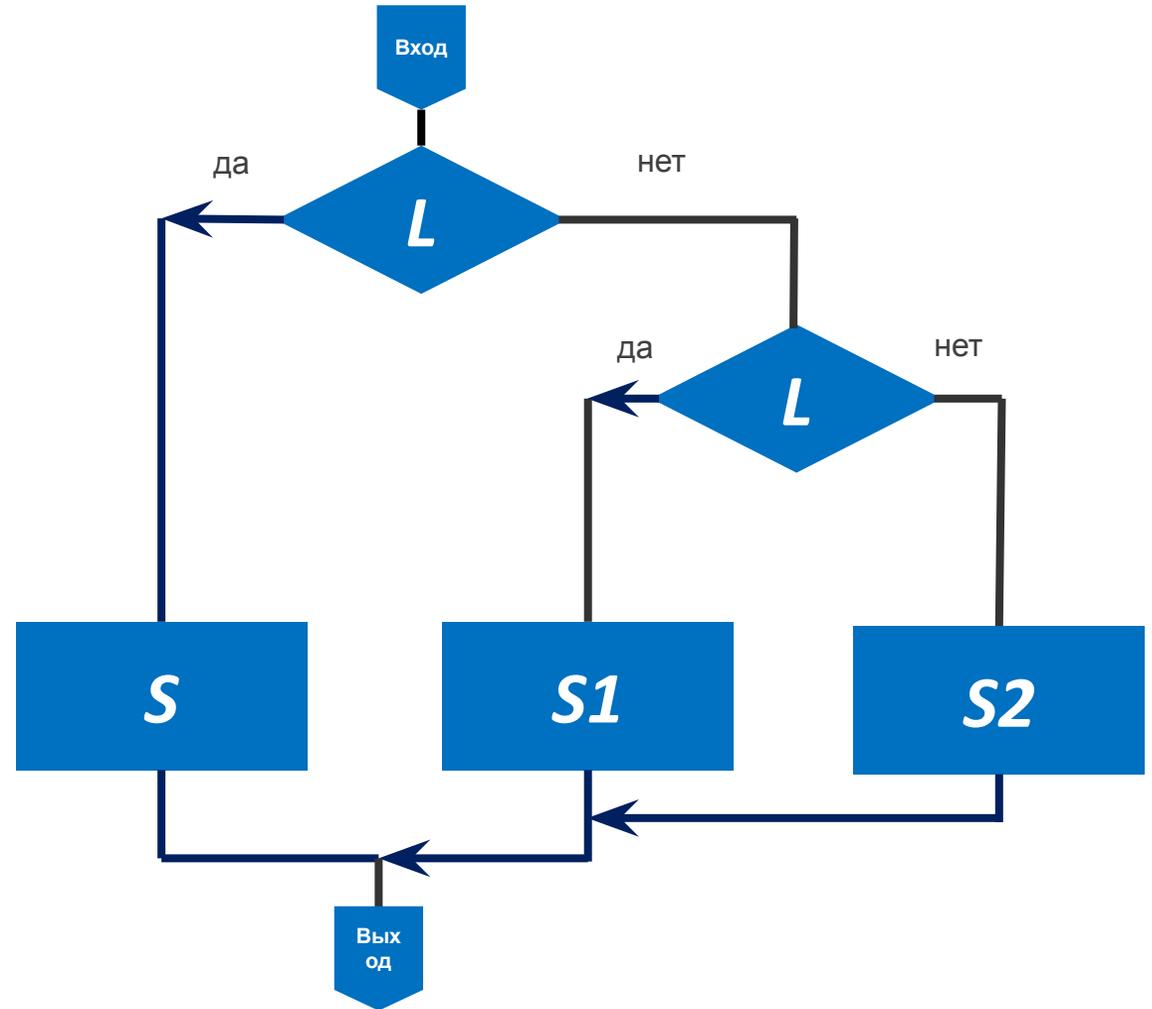
```
IF L THEN
  S
ELSE
  IF L1 THEN
    S1
  ELSE
    S2
  END IF
END IF
```



Двойное ветвление

```
ИЛИ  
IF L THEN  
  S  
ELSE IF L1 THEN  
  S1  
ELSE  
  S2  
END IF
```

Второй оператор короче на один END IF, но не так нагляден, как левый, в котором каждый IF заканчивается собственным END IF



Упрощенная развилки

Структура упрощенной развилки показана справа.

В отличие от структуры «двойное ветвление» эта структура содержит не два действия, а одно (S), и это действие либо выполняется (если условие L истинно), либо пропускается (если L ложно). По этой причине структуру такого типа иногда называют структурой «**обход**».

Строчный оператор для этой структуры имеет вид

```
IF L THEN S,
```

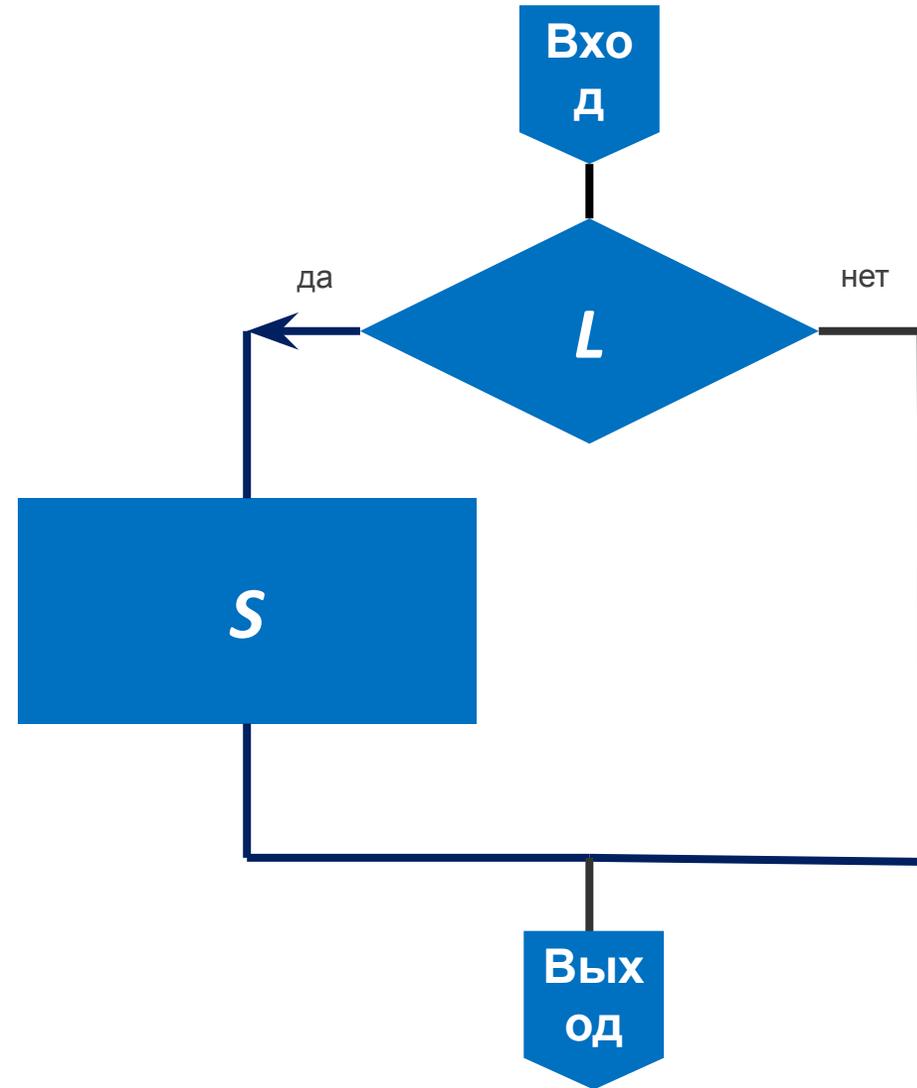
а блочный оператор

```
IF L THEN
```

```
  S
```

```
END IF
```

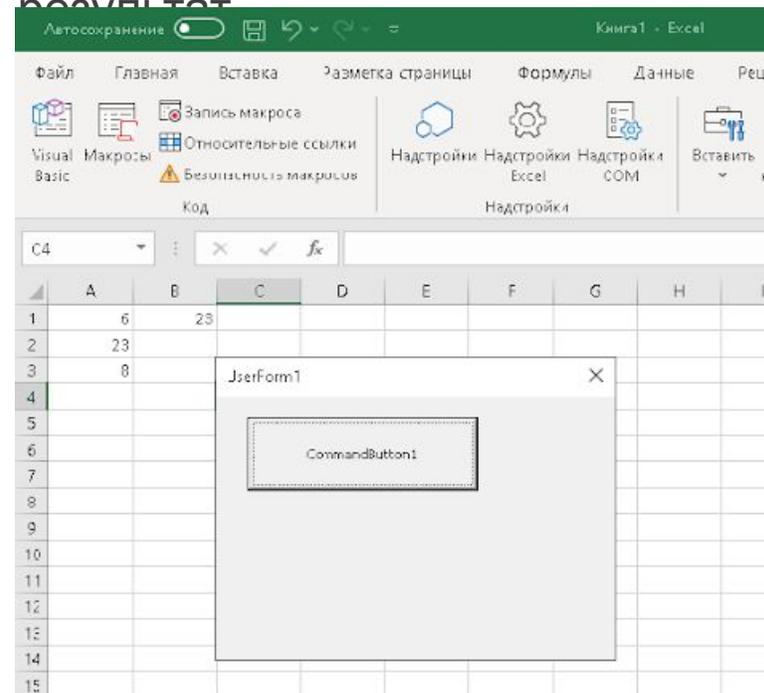
Структуры типа «обход» также допускают вложение друг в друга. В блочные операторы можно вкладывать строчные, но при этом надо обеспечивать правильность логики сложных конструкций, и с этой точки зрения наличие ключевых слов END IF позволяет успешно решить эту задачу. Чтобы при программировании не возникало логических противоречий, надо помнить, что ELSE всегда назначается на ближайший IF.



Пример программы на VBA

- Рассмотрим следующий пример. Пусть на листе Excel в ячейке A1, A2 и A3 введены три различных числа. Требуется в ячейку B1 поместить наибольшее из них.
- Напишем на VBA следующий код
- Private Sub CommandButton1_Click()
• If Range ("A1" »).Value > Range ("A2").Value Then
 - If Range("A1").Value > Range("A3").Value Then
 - Range("B1").Value = Range("A1").Value
 - Else
 - Range("B1").Value = Range("A3").Value
 - End If
- Else
- IF Range("A2").Value > Range ("A3").Value Then
- Range("B1").Value = Range ("A2").Value
- Else
- Range ("B1").Value =Range("A3").Value
- End If
- End If
- End Sub

Откроем редактор VBA, добавим форму UserForm1, поместим на эту форму командную кнопку CommandButton1, дважды щелкаем на ней и наберем в событийной процедуре этой кнопки данный код. После этого с помощью команды RUN запустим процедуру на выполнение и получим





Циклические структуры

Циклом называется алгоритм, обеспечивающий повторение заданного набора операторов. Циклы являются фундаментальной частью любого языка программирования. Практически ни одна программа для решения прикладных задач не может быть создана без использования циклов. Все циклы можно разделить на две группы: циклы, в которых число повторений известно до начала выполнения цикла, - это конструкция *For ... next*, и циклы, в которых число повторений определяется в процессе выполнения цикла (обычно для этого задаются какие-то условия).

Циклические структуры

К таким циклам относятся следующие конструкции

Do While условие

Операторы

Loop

Do

Операторы

Loop While условие

Do Until условие

Операторы

Loop

Do

Операторы

Loop Until условие

На VBA реализована также и конструкции цикла для работа с объектами, которая выполняется только в объектно-ориентированных языках:

For Each ... Next

В циклах типов «**Do While** условие» и «**Do Until** условие» условие окончания цикла проверяется до начала выполнения операторов цикла, и такие циклы, вообще говоря, могут не выполняться ни разу. Например, если цикл предназначен для чтения записей файла, а в файле нет записей, то условие окончания цикла это обнаружит и цикл не будет выполняться.

В двух других конструкциях условие проверяется в конце, и такой цикл обязательно должен быть выполнен по крайней мере один раз, поэтому его нежелательно применять для операции чтения файлов.

Рассмотрим пример использования циклов.

Примеры программы с циклами на VBA

- Пусть на Лист 1 (Sheet1) записана таблица, в которой указан средний балл студентов. Требуется окрасить светло-зеленым цветом те строки таблицы, в которых средний балл больше 4. Для этого напишем для процедуры элемента CommandButton1 следующий код:
- Private Sub CommandButton1_Click()
- Dim I, n As Integer
- n=Cint (InputBox("n="))
- For i=2 to n
- If Cells(i,4)>=4 then
- Cells(i,1).Resize(1,4).Interior.ColorIndex = 35
- End If
- Next
- End Sub

Результат выполнения показан на рисунке справа

The screenshot shows the Excel interface with the 'Разработка' (Developer) ribbon active. A UserForm titled 'UserForm1' is displayed over the spreadsheet, containing a single CommandButton1. The spreadsheet data is as follows:

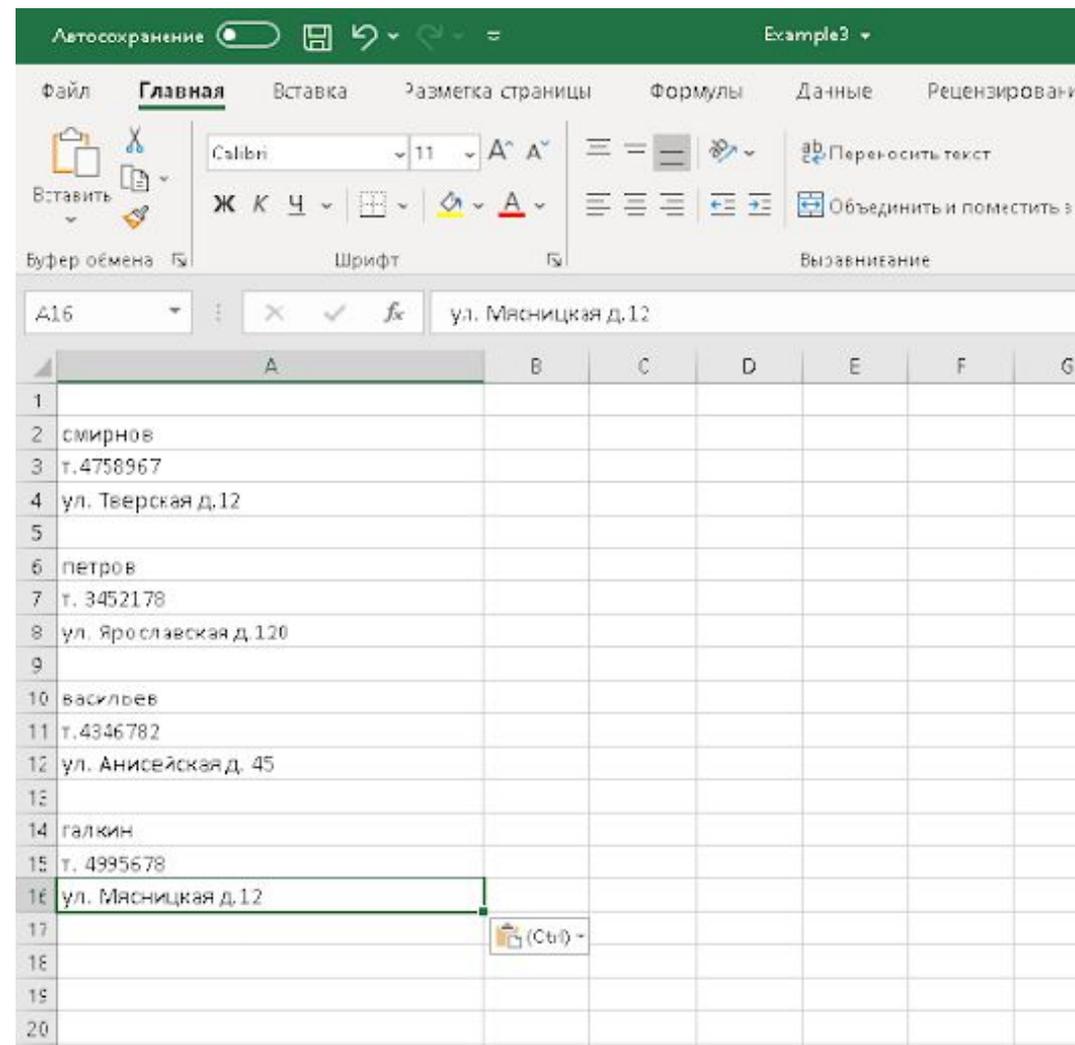
	A	B	C	D	E	F	G	H	I	J	K
1	Фамилия	Группа	Курс	Средний балл							
2	Смирнов	1-11	1	3,34							
3	Петров	1-11	1	3,80							
4	Галкин	1-10	1	4,23							
5	Маркина	1-10	1	3,91							
6	Морозова	2-17	2	4,87							
7	Волков	3-1	3	3,11							
8	Еремин	3-2	3	4,27							
9	Зипин	3-11	3	3,25							
10											
11											
12											
13											
14											
15											
16											
17											

Примеры программы с циклами на VBA

Циклы *Do ... Loop* используются в прикладных задачах очень часто. Рассмотрим такой пример. Пусть по электронной почте были получены данные о некоторых клиентах. Эти данные определенным образом размещены в первом столбце MS Excel.

Требуется переупорядочить эти данные, разместив их, как в базе данных: в столбце A должна находиться фамилия, в столбце B – номер телефона и в столбце C – адрес.

При написании программы на VBA можно часть кода для пересылки значений из одной ячейки в другую создать с помощью функции Record Macro, установив стиль относительных ссылок (Use Relative Reference). При этом надо отформатировать ячейки как текст, увеличив их ширину и сделать активной ячейку A1. Затем войти в редактор кода и к имеющемуся там коду добавить цикл Do Until ... Loop. После этого запустить Макрос Macro1.



Примеры программы с циклами на VBA

```
Sub Macro1()
```

```
Do Until Not Selection.Value > ""
```

```
    ActiveCell.Offset(1, 0).Range("A1").Select
```

```
    Selection.Cut
```

```
    ActiveCell.Offset(-1, 1).Range("A1").Select
```

```
    ActiveSheet.Paste
```

```
    ActiveCell.Offset(2, -1).Range("A1").Select
```

```
    Selection.Cut
```

```
    ActiveCell.Offset(-2, 2).Range("A1").Select
```

```
    ActiveSheet.Paste
```

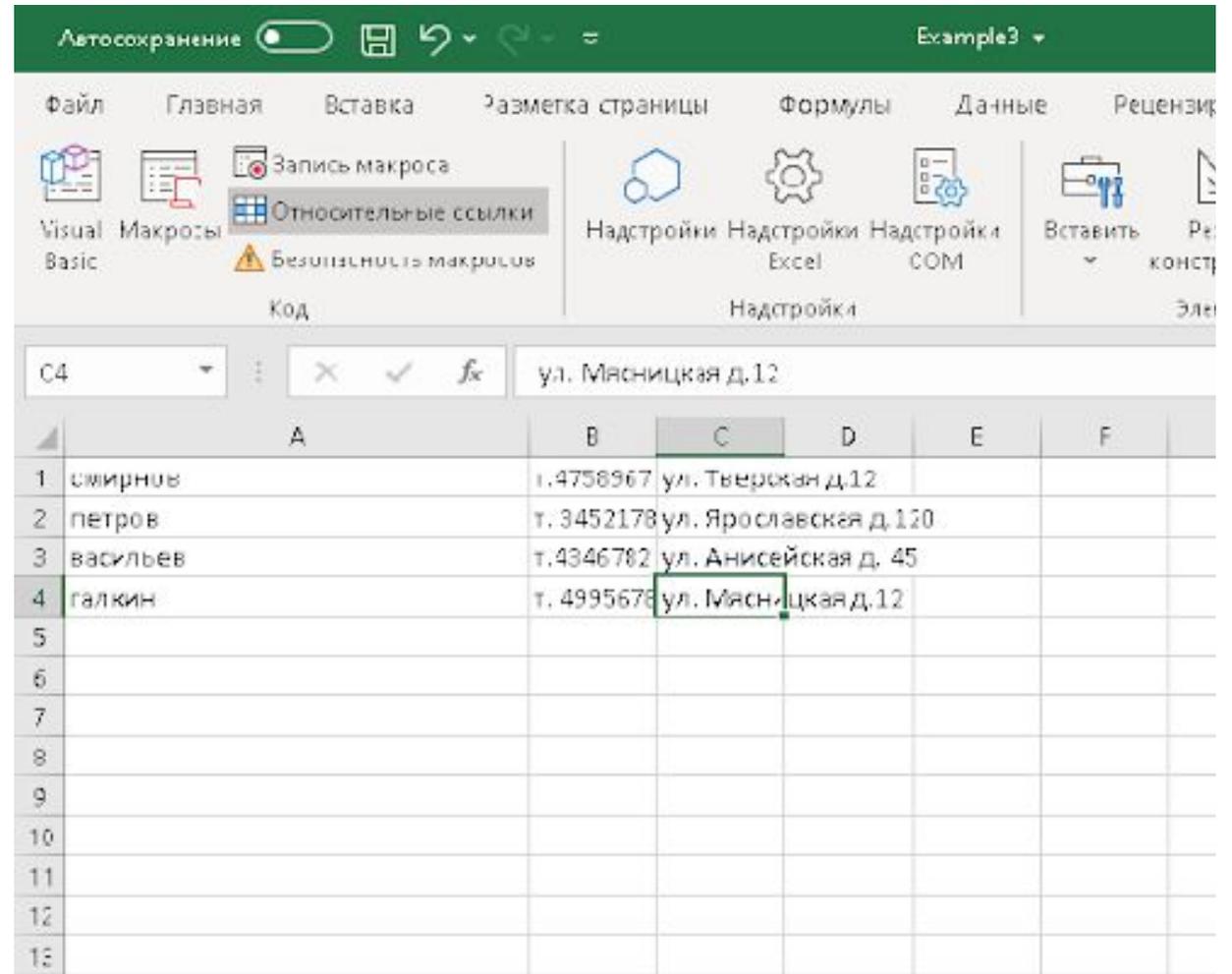
```
    ActiveCell.Offset(1, -2).Range("A1:A3").Select
```

```
    Selection.EntireRow.Delete
```

```
    ActiveCell.Select
```

```
Loop
```

```
End Sub
```



Контрольные вопросы и упражнения

Найдите в Интернете и дополнительной литературе ответы на следующие вопросы:

1. **Какие типы данных поддерживаются в VBA?**
2. **Каковы правила образования имен?**
3. **Каков синтаксис описания констант?**
4. **Какие виды констант существуют?**
5. **Каков синтаксис описания переменных?**
6. **Как представляются числа в VBA?**
7. **Что такое арифметические типы и выражения?**
8. **Что такое логические типы и выражения?**
9. **Что такое символические типы и выражения?**
10. **Каков смысл и синтаксис оператора Like?**
11. **Каково назначение и виды процедур и функций?**
12. **Каков синтаксис объявления процедур?**
13. **Каков синтаксис объявлений функций?**
14. **Приведите стандартные математические функции?**
15. **Приведите стандартные функции преобразования**
16. **Приведите стандартные строчные**
17. **Каковы назначение и синтаксис встроенного диалогового окна InputBox?**
18. **Каковы назначение и синтаксис встроенного диалогового окна MsgBox?**
19. **Какие условные операторы используются в VBA?**
20. **Каковы назначения и синтаксис оператора варианта Select Case?**
21. **Каковы назначение и синтаксис конструкции «цикл с параметром»?**
22. **Каковы назначение и синтаксис конструкций цикла с предусловием, цикла с постусловием?**
23. **Какие существуют разновидности массивов и способы работы с ними?**

Упражнение. Создать процедуру, которая вычисляет сумму и произведение цифр заданного трехзначного числа.