

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования  
**«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ИНДУСТРИАЛЬНЫЙ  
УНИВЕРСИТЕТ»**

Кафедра «Информационных технологий и систем  
в экономике и управлении»

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**  
**по выполнению по курсовой работы по дисциплине**  
**«Структуры и алгоритмы компьютерной обработки данных»**

Составитель  
Н.Н. Карабутов

Москва 2013

## Введение

Методические указания предназначены для выполнения курсовой работы по дисциплине «Структуры и алгоритмы компьютерной обработки данных».

**Целью курсовой работы** является освоение методов работы со структурами и алгоритмами компьютерной обработки данных и их применение для решения практических задач.

**Задачей курсовой работы** является изучение методов и алгоритмов компьютерной обработки данных.

### 1. Задание на курсовую работу по САКОД

Разработать программу на VBA в Excel для ввода данных и анализа списка (базы данных).

База данных имеет вид (табл. 1)

Таблица 1

№ пп	Город	Товар	Время доставки	Затраты

Таблица должна содержать не менее 10 записей (строк).

Для создания таблицы использовать тип данных **Type**.

**Примечание 1.** Описание типа данных **Type** и создаются в модуле..

*Алгоритм работы программы.*

1. На листе 1 создать таблицу с данными и затем их считать и вывести на Листе 2 или Листе 1.

**Примечание 2.** Можно предусмотреть непосредственный ввод данных в программе.

2. Применяя рекурсию, найти суммарные и средние затраты по всем товарам. Результаты вывести на лист.

**Примечание 3.** Рекурсию реализовать в виде процедуры – функции.

3. Отсортировать выведенную таблицу по полю время доставки по любому критерию и результат вывести на лист.

4. Используя команду Excel SUM (сумма) в созданной процедуре, проверить правильность полученных суммарных и средних значений.

**Варианты заданий.**

1. База данных имеет вид

№ пп	Производитель товара	Покупатель	Товар	Стоимость	Затраты на доставку

1.1. Определить суммарную стоимость товара, используя рекурсию.

1.2. Отсортировать таблицу по полю Товар.

1.3. Используя команду Excel AVERAGE (среднее) в созданной процедуре, проверить правильность полученных суммарных и средних значений.

1.4. Определить суммарные затраты на доставку продукции для каждого Производитель товара.

### 2. Порядок выполнения работы

Для выполнения курсовой работы необходимо разработать алгоритм обработки данных, процедуры его реализации в Excel с помощью языка VBA.

## 2.1. Создание процедуры

Процесс создания процедуры (макроса) в Excel сводится к реализации следующих действий.

1. Перейти на лист, например 1\_массив, содержащий данные, для обработки которых создается макрос, и нажать комбинацию клавиш **ALT+F11**. В результате на экране откроется приложение VBA.

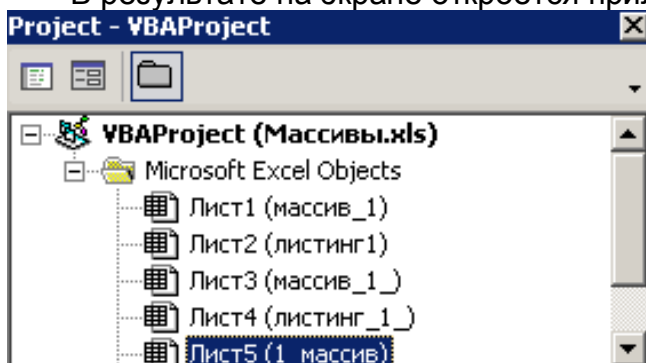


Рис. 2. Панель проектов

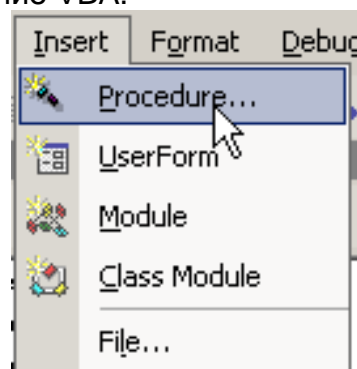


Рис. 3. Раздел меню Insert

2. Для вставки подпрограммы-процедуры необходимо:

- а) в окне проектов (рис. 2) выбрать лист 1\_массив, в котором должен создаваться макрос;
- б) применить команду **Procedure** из меню **Insert** (рис. 3);
- в) в появившемся на экране диалоговом окне **Add Procedure** (*Добавить процедуру*) задать имя Массив1 процедуры (рис. 4);
- г) нажать кнопку **OK**.

В результате в окне создания кода появится заготовка подпрограммы:

```
Sub Массив1()
```

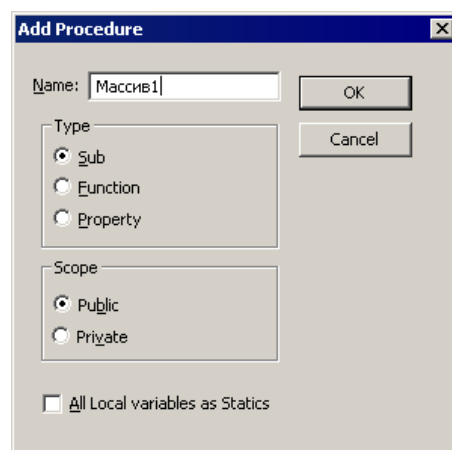



Рис. 4. Диалоговое окно Add Procedure

```
End Sub
```

3. Написать код для обработки данных. Пример кода:

```
Sub Массив1()  
    'Описание массива из 10 элементов  
    Dim x(1 To 10) As Integer,  
    For j = 1 To 10  
        x(j) = j * j  
    Next j  
End Sub
```

4. Для проверки работоспособности работы процедуры Массив1 нажать F5 или кнопку  **Запустить подпрограмму** на панели инструментов.

## 2.2. Работа с электронной таблицей

Итак, после знакомства с элементами языка VBA и основными математическими, текстовыми и диалоговыми функциями перейдем непосредственно к созданию макросов в Excel. Сначала рассмотрим объекты, а точнее, свойства листа в Excel, позволяющие

вносить результаты вычислений в ячейки электронной таблицы. Здесь могут применяться следующие два подхода.

1. Если используется стандартная адресация ячеек типа *столбец - строка*, т. е. например, A5, то для записи в ячейке значения используется объект **Range** рабочей книги, который может представлять собой ячейку, строку, столбец, диапазон ячеек. Объект имеет некоторую совокупность свойств. Он имеет следующий синтаксис

Синтаксис

**Range(Cell1)** или **Range(Cell1, Cell2)**

где Cell1 — адрес ячейки.

**Пример.**

а) **Range("A5")=5**

ячейке A5 присваивается значение 5;

б) **Range("A5:D8")=45**

диапазон ячеек A5:D8 заполняется значениями 45.

**Замечание.** Адрес в объекте **Range** задается в виде строки символов.

2. Если используется адресация ячеек типа *строка - столбец*, то вместо адреса A5 в объекте **Range** рабочей книги необходимо использовать свойство Cells(5, 1). Тогда пример а) можно записать следующим образом

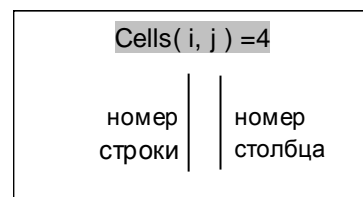
а) **Range(Cells(5,1)) =5.**

Для представления диапазона ячеек (пример б) используется вторая форма объекта **Range**. Для примера б) имеем:

**Range(Cells(5,1), Cells(8,4))=45**

Для присвоения значения отдельной ячейке используется свойство Cells. Например, **Range("A5")=5** можно записать

**Cells(5,1)= 5.**



### 2.3. Создание модуля в VBA

Процесс создания модуля аналогичен созданию процедуры, только при работе с VBA необходимо сначала войти в меню **Insert** (рис. 3) и выбрать раздел **Module**. В результате в окне проектов появится раздел Modules (рис. 5).

Теперь можно создавать процедуры для реализации алгоритмов указанных в задании.

Пример применения инструкции **Type** для описания структуры базы данных показан ниже. База состоит из 2 полей №пп и Товар.

В верхней части описана структура базы. Далее приведена процедура Список, которая уже работает с конкретной базой данных, имеющей тип N1/

Option Explicit

Type N1 ' Тип, определенный пользователем.

ID As Integer ' Элементы типа данных № пп.

N As String ' Элементы типа данных Товар

End Type

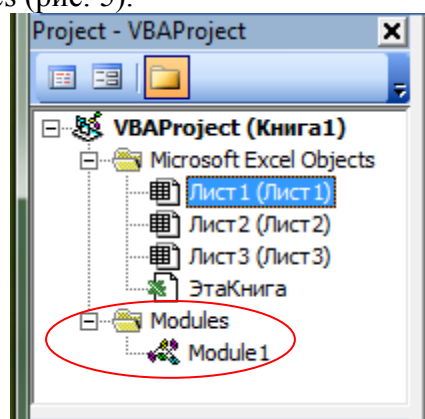


Рис. 5

```

Sub Список()
Dim a(1 To 4) As N1, i%
For i = 1 To 4
    a(i).ID = Cells(2 + i, 1)
    a(i).N = Cells(2 + i, 2)
Next
For i = 1 To 4
    Cells(2 + i, 5).Font.ColorIndex = 5
    Cells(2 + i, 5) = a(i).N
    Cells(2 + i, 4).Font.ColorIndex = 3
    Cells(2 + i, 4) = a(i).ID
Next
End Sub

```

Пример работы процедуры Список показан на рис. 6.

	A	B	C	D	E	F
1	Исходный список					
2	№ пп	Товар		Вывод результатов		
3	1	Станки		1	Станки	
4	2	Сапоги		2	Сапоги	
5	3	Порошок стиральный		3	Порошок стиральный	
6	4	Паста		4	Паста	
7						
8						
9						
10	Работает только на уровне модуля.					
11	Чтобы вывести на лист, следует сначала активировать лист					
12	затем Модуль-> Процедуру и F5					
13						

Рис. 6

Пример рекурсивной процедуры, позволяющей получить произведение чисел от 1 до до 7:

	A	B	C	D	E	F	G
1	1	2	6	24	120	720	5040
2							

```

Function factorial(N As Integer) As Integer

```

```

If N = 0 Then

```

```

    factorial = 1

```

```

Else

```

```

    factorial = N * factorial(N - 1)

```

```

End If

```

```

End Function

```

```

Public Sub asd()

```

```

Dim y%, N%, i%

```

```

N = 7

```

```

For i = 1 To N

```

```

    y = factorial(i)

```

```

    Cells(1, i) = y

```

```

Next i

```

```

End Sub

```

### 3. Представление результатов

Курсовая работа должна иметь следующую структуру.

1. Задание с заполненной исходной базой данных (табл. 1).

2. Описание структуры базы данных с помощью инструкции Type.
3. Алгоритм работы процедуры работы с базы данных.
4. Листинг процедуры, реализующей алгоритм.
5. Реализацию рекурсивного алгоритма в виде процедуры.
6. Численные результаты работы алгоритмов обработки данных.
7. Используемая литература.