Федеральное агентство железнодорожного транспорта

Сибирский государственный университет путей сообщения

Кафедра "Общая информатика"

КУРСОВАЯ РАБОТА

по дисциплине "Информатика"

РЕШЕНИЕ ЭКОНОМИЧЕСКИХ И БУХГАЛТЕРСКИХ ЗАДАЧ

С ИСПОЛЬЗОВАНИЕМ ИНСТРУМЕНТАРИЯ

VISUAL BASIC FOR APPLICATION

Руководитель: Разработал

ст. преподаватель студент гр. БА-111

Макарова Е.А.

*Краткая рецензия:* Парфенович К.И.

*Новосибирск 2009 г.*

Задание на выполнение работы для студентов 1 курса

Группа-БА-111

Ф. И.О. студента Парфенович Ксения Игоревна. Шифр студента 13-08

*Исходные данные***:**

num-номер группы;

n5 - количество "5";

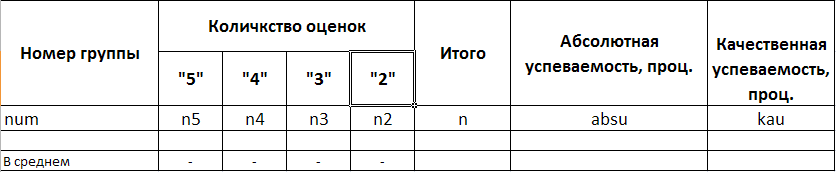
n4 - количество "4";

n3-количество "3";

n2-количество "2".

*Вычислить***:** n=n2+n3+n4+n5; absu= (n3+n4+n5) /n\*100; kau= (n4+n5) /n\*100

*Результаты представить в следующем виде:*



Количество строк n=9

Количество строк дозаписи исходных данных k=4

Сортировка даных по Абсолютная успеваемость absu

Выборка данных по Качественная успеваемость kau

Определение максимального и минимального значения

Создание диаграмм: гистограмма - absu и kau; круговая диаграмма - absu, kau, смешанная - absu и kau

Задание выдано 03.04.09

/Макарова Е.А. /

## Аннотация

Данная работа посвящена решению экономических задач с использованием Microsoft Excel и инструментария Visual Basic For Application.

Курсовая работа состоит из:

34 страниц;

12 рисунков;

8 программ;

1 таблицы;

Содержание

Введение

1. Способы запуска редактора Visual Basic

2. Главные правила синтаксиса VBA

3. Типы данных

4. Исходные данные

5. Копирование данных с использованием процедур в программах

6. Дозапись исходных данных

7. Создание отчёта

8. Сортировка данных

9. Создание автоматического макроса по сортировке

10. Выборка данных

11. Создание автоматического макроса по выборке

12. Определение максимального и минимального значения

13. Создание диаграмм (гистограмма, круговая, смешанная)

14. Создание управляющих кнопок

15. Автоматический макрос Гашение

Заключение

Список использованной литературы

## Введение

Информатика это предмет, изучающий компьютер и его составляющие, а также различные программы позволяющие производить расчеты, делать графики составлять таблицы и др. Одной из программ позволяющих нам составлять таблицы и производить расчеты и работающий в среде Windows является Microsoft Excel. Visual Basic используется в качестве внутреннего языка программного пакета Microsoft Office (Word, Excel и др.). С помощью Visual Basic можно создавать приложения практически для любой области современных компьютерных технологий: бизнес-приложения, игры, мультимедиа, базы данных.

Сейчас этот язык используют не только программисты-любители, но и все больше коммерческие организации. Так, например абсолютное большинство организаций занимающихся реализацией компьютерной техники используют в своей работе прайс-листы, созданные в Excel, с обработкой данных программами написанными на языке Visual Basic.

Visual Basic является мощным программным средством, с помощью которого можно реализовать широкий спектр практических задач. *Visual Basic* - это объектно-ориентированный язык. Основой языка являются объекты.

## 1. Способы запуска редактора Visual Basic

Прежде чем начать работать с редактором Visual Basic, нужно его открыть. Во всех приложениях Office это делается одинаково:

самый простой способ: в меню Сервис выбрать Макрос - > Редактор Visual Basic;

самый быстрый способ: нажать <Alt>+<F11>;

можно также воспользоваться кнопкой на панели инструментов Visual Basic (предварительно сделав ее видимой);

можно вызвать редактор при возникновении ошибки в макросе;

можно открыть готовый макрос на редактирование в диалоговом окне Макрос.

В любом случае откроется окно, похожее на представленное на рис.

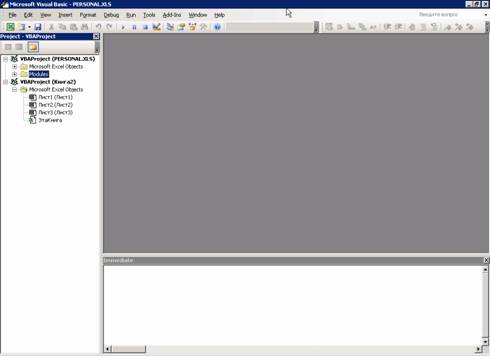


Рис.1 Окно редактора Visual Basic в Excel

## 2. Главные правила синтаксиса VBA

Синтаксис VBA, как понятно из самого названия этого языка (которое расшифровывается как Visual Basic for Applications), почти полностью совпадает с синтаксисом Visual Basic. Некоторые основные синтаксические принципы этого языка:

VBA нечувствителен к регистру;

чтобы закомментировать код до конца строки, используется одинарная кавычка (') или команда REM;

символьные значения должны заключаться в двойные кавычки;

максимальная длина любого имени в VBA (переменные, константы, процедуры) - 255 символов;

начало нового оператора - перевод на новую строку;

ограничений на максимальную длину строки нет (хотя в редакторе умещается только 308 символов).

## 3. Типы данных

Тип данных определяет, каким образом биты данных, представляющие конкретное значение, хранятся в памяти ПК. В каждом языке программирования имеется свой фиксированный набор базовых типов данных. Некоторые языки позволяют создание дополнительных (пользовательских) типов данных. В VBA имеются следующие типы данных:

|  |  |  |
| --- | --- | --- |
| Тип данных | Размер | Диапазон значений |
| Byte (байт) | 1 байт | От 0 до 255 |
| Boolean (логический) | 2 байт | True или False. |
| Integer (целое) 2 байт | 2 байт | От - 32 768 до 32 767. |
| Long (длинное целое) | 4 байт | От - 2 147 483 648 до 2 147 483 647 |
| Single (с плавающей точкой обычной точности) | 4 байт | От - 3,402823E38 до - 1,401298E-45 для отрицательных значений; от 1,401298E-45 до 3,402823E38 для положительных значений. |
| Double (с плавающей точкой двойной точности) | 8 байт | От - 1,79769313486232E308 до - 4,94065645841247E-324 для отрицательных значений; от 4,94065645841247E-324 до 1,79769313486232E308 для положительных значений |
| Currency (денежный) | 8 байт | От - 922 337 203 685 477,5808 до 922 337 203 685 477,5807 |
| Decimal (масштабируемое целое) | 14 байт | +/-79 228 162 514 264 337 593 543 950 335 без дробной части; +/-7,9228162514264337593543950335 с 28 знаками справа от запятой; минимальное ненулевое значение имеет вид +/-0,0000000000000000000000000001. |
| Date (даты и время) | 8 байт | 1 января 100 г. до 31 декабря 9999 г |
| Object (объект) | 4 байт | Любой указатель объекта |
| String (строка переменной длины) | 10 байт+ длина строки | От 0 до приблизительно 2 млрд. |
| String (строка постоянной длины) | Длина строки | От 1 до приблизительно 65 400 |
| Variant (числовые подтипы) | 16 байт | Любое числовое значение вплоть до границ диапазона для типа Double |
| Variant (строковые подтипы) | 22 байт + длина строки | Как для строки (String) переменной длины |

## 4. Исходные данные

На Листе 2 создается таблица 1 путем набора исходных данных согласно заданию. Программным путем определяется количество строк и столбцов таблицы с помощью функции Empty.

Данная функция проверяет строки по выбранному столбцу (или строке). Если строка не пустая, то подсчитывается количество строк таблицы, передача управления на проверку следующей строки выполняется с помощью оператора безусловной передачи Go To на метку М1.

Если строка пустая, то выполняется передача управления с помощью оператора Go To на метку М2, где фиксируется количество заполненных строк таблицы. В конце метки заканчивается двоеточием.

*Sub подсчет\_строк\_столбцов ()*

*Sheets ("Лист2"). Select*

*i = 1 'формирование строки таблицы*

*M1:*

*Stroka = Sheets ("Лист2"). Cells (i + 5,1) 'анализируемая ячейка*

*If Stroka = Emty Then 'если строка пустая*

*GoTo M2 'переход на метку 2*

*Else*

*i = i + 1 'подсчет количества строк*

*GoTo M1 ' переход к проверке следующей строки*

*End If*

*M2: 'в том случае, если ячейка пустая*

*n = i - 1 'подсчет количества строк в таблице*

*'n-количество строк в таблице*

*Sheets ("Лист2"). Cells (5,11) = n*

*j = 1 'формирование столбца таблицы*

*M3:*

*Stolbets = Sheets ("Лист2"). Cells (4, j) 'анализируемый столбец*

*If Stolbets = Emty Then 'если столбец пустой*

*GoTo M4 'переход на метку 4*

*Else*

*j = j + 1 'подсчет количества столбцов*

*GoTo M3 ' переход к проверке следующего столбца*

*End If*

*M4: 'в том случае, если столбец пустой*

*m = j - 1 'подсчет количества столбцов в таблице*

*'m-количество столбцов в таблице*

*Sheets ("Лист2"). Cells (5,12) = m*

*End Sub*



Рис.2 Данные подсчёта количества строк и столбцов

## 5. Копирование данных с использованием процедур в программах

Скопировать исходные данные с Листа2 программным путём, применяя программы процедур ввода и вывода данных.

Процедуры - это самые важные функциональные блоки языка VBA. В VBA вы можете выполнить тот программный код, который содержится в какой-либо процедуре.

Макрос в VBA - это процедура типа Sub, не имеющая параметров. Только макросы можно вызвать по имени из редактора VBA или из приложения Office.

Процедуры условно делятся на процедуры - подпрограмм и процедуры-функции.

Отличаются они тем, что процедура - подпрограмм просто выполняет группу операторов, а функция вычисляет некоторые значения и передаёт его обратно в главную программу.

Чтобы работа подпрограмм имела смысл, её надо получить данные из главной (вызывающей) программы (главного модуля), которая эту программу вызывает. Данные передаются подпрограмме в виде параметров или аргументов, которые обычно описываются в её заголовке так же, как и переменные. Подпрограммы активизируются только в момент их вызова. Операторы, находящиеся в нутрии программы, выполняются только в том случае, если эта подпрограмма явно вызвана. Пока выполнение программы полностью не закончится, оператор главной программы, следующей за командой вызова подпрограммы, выполнятся не будет.

Синтаксис главного модуля:

*Sub < Имя программы> ()*

*< объявление данных>*

*< имя процедуры> (список фактических парметров)*

*End Sub*

*Синтаксис процедуры:*

*Sub <имя программы> (список фактических параметров)*

*Операторы*

*End Sub*

Создаётся главный модуль, в котором указывается имена подпрограмм и фактических параметров (лист, имя массива, размерность массива и т.д.). Управления работой процедурами выполняется из главного модуля.

Каждая процедура начинается с оператора Sub - начало программы, имени, отмечающее её начало, перечисляются аргументы, которые передаются процедуре при вызове. С момента и до окончания работы вызываемой процедуры устанавливается соответствие между списком фактических и формальных параметров, поэтому списки фактических и формальных параметров должны иметь одинаковый смысл.

*'главная программа*

*Sub копирование ()*

*Sheets ("Лист3"). Select*

*Dim A () As Variant, m, n As Integer 'объявление динамического двумерного массива*

*'обращение к подпрограмме ввода данных в память*

*n = Sheets ("Лист2"). Cells (5,11) ' кол-во строк*

*m = Sheets ("Лист2"). Cells (5,12) ' кол-во столбцов*

*ReDim A (1 To n, 1 To m) ' переобъявление массива*

*'процедура ввода*

*VVOD "Лист2", A, n, m, 3 'список фактических параметров, где:*

*'Лист2-лист, с которого данные вводятся в память;*

*'а-имя массива;*

*'n-количество строк массива;*

*'m-количество столбцов массива;*

*'3-количество строк заголовка таблицы.*

*'обращение к подпрограмме вывода данных на лист*

*VIVOD "Лист3", A, n, m, 3 'вывод из памяти на Л3*

*VIVOD "Лист4", A, n, m, 3 'вывод из памяти на Л4*

*End Sub*

*'Подпрограмма "VVOD":*

*Sub VVOD (L, x, y, r, S) 'список фактических пареметров, где:*

*'L-лист;*

*'x-имя массива;*

*'y-количество строк в массиве;*

*'r-количество столбцов в массиве;*

*'S-колчество строк заголовков таблицы*

*For i = 1 To y*

*For j = 1 To r*

*x (i, j) = Sheets (L). Cells (i + S, j)*

*Next j*

*Next i*

*End Sub*

*'Подпрограмма "VIVOD":*

*Sub VIVOD (L, x, y, r, S)*

*For i = 1 To y*

*For j = 1 To r*

*Sheets (L). Cells (i + S, j) = x (i, j)*

*Next j*

*Next i*

*End Sub*



Рис.3 Программа применения процедур ввода и вывода

## 6. Дозапись исходных данных

Для реализации действий, используется оператор Slect Case.

Синтаксис:

*Slect Case < проверяемое выражение>*

*Case < значении, список значений, диапазон значений>*

*……………………*

*<инструкции>*

*Case < значении, список значений, диапазон значений>*

*<инструкции>*

*……………………*

*<инструкции>*

*Case < значении, список значений, диапазон значений>*

*<инструкции>*

*……………………*

C Листа2 копируем исходные данные на Лист4 (процедурами ввода данных в память и вывода на лист). Объявляем матрицу как динамичесий массив. Считываем с Листа 2 количество строк (n) и столбцов (m), размерность которой будет n\*m. Вводим новую матрицу в память, затем на лист. Формируем новую матрицу, чтобы выполнить дозапись. Ввод данных производится через оператор InputBox, ввод новых элементов выполняется в цикле For. Снова переобъявляем матрицу, размерность которой уже будет (n+k) \*m. Выводим на лист новую матрицу.

*Sub Дозапись ()*

*Sheets ("Лист4"). Select*

*Dim A () As Variant 'объявление динамического двумерного массива*

*'обращение к подпрограмме ввода данных в память*

*n = Sheets ("Лист2"). Cells (5,11) 'количество строк массива*

*m = 5 'количество столбцов массива*

*ReDim A (1 To n, 1 To m)*

*k = InputBox ("Введите количество дозаписываемых элементов")*

*Sheets ("Лист4"). Cells (5, 13) = k 'количество дозаписываемых элементов*

*ReDim A (1 To n + k, 1 To m) As Variant*

*For i = 1 To n*

*For j = 1 To m 'столбцы для дозаписи даннных*

*A (i, j) = Sheets ("Лист4"). Cells (i + 2, j)*

*Next j*

*Next i*

*'вывод новых элементов на Л4*

*For i = 1 To k*

*For j = 1 To m*

*If j = 1 Then 'номера столбцов, по которым вводятся данные*

*'y указывает наименование переменной вводимого столбца*

*y = "Номер группы"*

*Else*

*If j = 2 Then*

*y = "Количество 5"*

*Else*

*If j = 3 Then*

*y = "Количество 4"*

*Else*

*If j = 4 Then*

*y = "Количество 3"*

*Else*

*If j = 5 Then*

*y = "Количество 2"*

*End If*

*End If*

*End If*

*End If*

*End If*

*A (i + n, j) = InputBox (y)*

*Sheets ("Лист4"). Cells (i + 4 + n, j) = A (i + n, j)*

*Next j*

*Next i*

*n1 = n + k*

*Sheets ("Лист4"). Cells (5,12) = n1*

*End Sub*

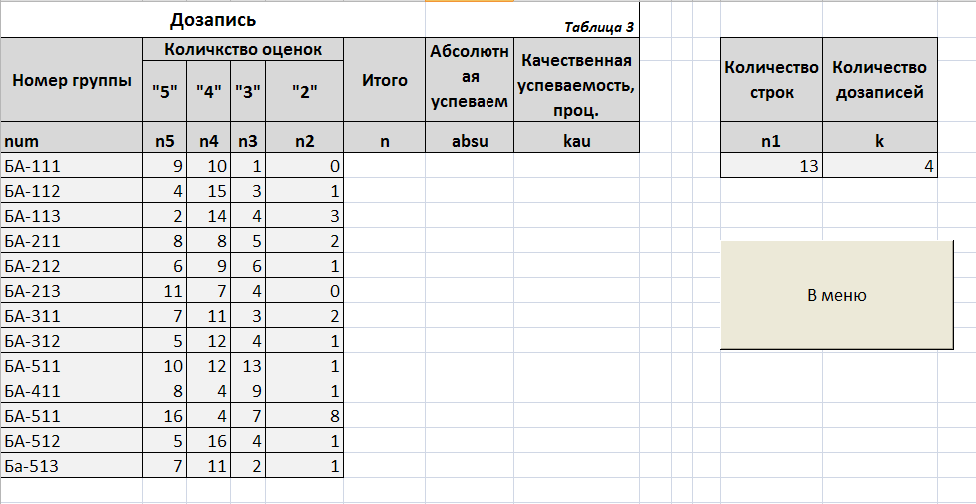


Рис.4. Данные программы после дозаписи

## 7. Создание отчёта

В отчете производим копирование данных с Листа 4 на Лист 5 и с помощью одномерного массива рассчитывается новые столбцы: "Итого", "Абсолютная успеваемость" и "Качественная успеваемость". Для нахождения производим расчет по формулам: n=n2+n3+n4+n5; absu= (n3+n4+n5) /n\*100; kau= (n4+n5) /n\*100. Производим считывание количества строк (n) и столбцов (m). Объявляем матрицу A как динамический массив через оператор ReDim. Копируем данные с листа 4 на лист 5 с помощью процедур. Затем высчитываем по формулам значения, а после находим среднее значение в столбцах "Итого", "Абсолютная успеваемость" и "Качественная успеваемость". И выводим на лист:

*Sub Отчет ()*

*Sheets ("Лист5"). Select*

*Dim A () As Variant*

*n1 = Sheets ("Лист4"). Cells (5,12) 'количество строк массива*

*M1 = Sheets ("Лист2"). Cells (5,12) 'количество столбцов массива*

*ReDim A (1 To n1, 1 To M1)*

*'Ввод данных в память c Листа4 и вывод на Лист5*

*VVOD "Лист4", A, n1, M1, 4*

*VIVOD "Лист5", A, n1, M1, 4*

*S = 0 ' S-ячейка для подсчета итоговой суммы по графе Средняя годовая стоимость имущества*

*'Составление отчета*

*For i = 1 To n1*

*For j = 1 To M1*

*A (i,*

*6) = A (i,*

*2) + A (i,*

*3) + A (i,*

*4) + A (i,*

*5)*

*Sheets ("Лист5"). Cells (i + 4,6) = A (i,*

*6)*

*Next j*

*S = S + A (i,*

*6)*

*Sr = S / n1*

*Sheets ("Лист5"). Cells (18,6) = Sr*

*Next i*

*S = 0*

*For i = 1 To n1*

*For j = 1 To M1*

*A (i,*

*7) = (A (i,*

*4) + A (i,*

*3) + A (i,*

*2)) / A (i,*

*6) \* 100*

*Sheets ("Лист5"). Cells (i + 4,7) = A (i,*

*7)*

*Next j*

*S = S + A (i,*

*7)*

*Sr = S / 13*

*Sheets ("Лист5"). Cells (18,7) = Sr*

*Next i*

*S = 0*

*For i = 1 To n1*

*For j = 1 To M1*

*A (i,*

*8) = (A (i,*

*3) + A (i,*

*2)) / A (i,*

*6) \* 100*

*Sheets ("Лист5"). Cells (i + 4,8) = A (i,*

*8)*

*Next j*

*S = S + A (i,*

*8)*

*Sr = S / n1*

*Sheets ("Лист5"). Cells (18,8) = Sr*

*Next i*

*Sheets ("Лист5"). Cells (18,1) = "В среднем"*

*End Sub*

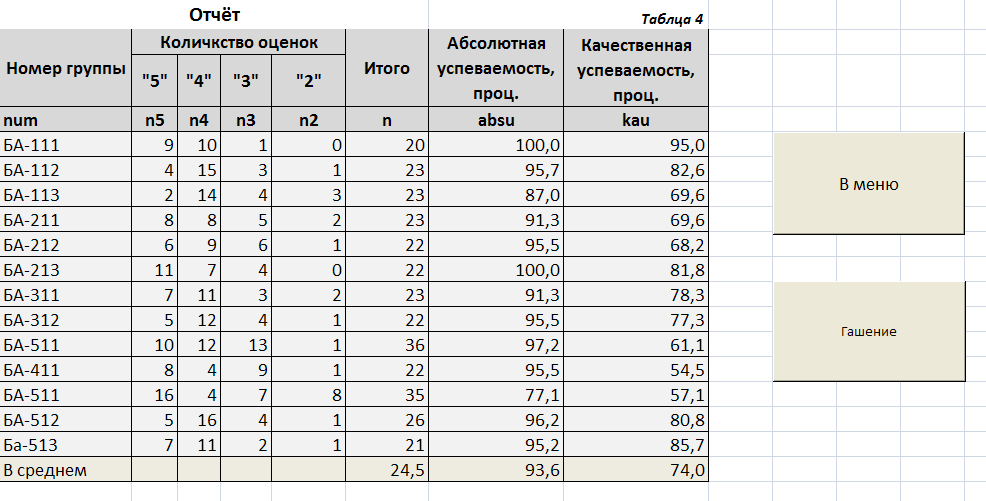


Рис.5. Данные таблицы после создания Отчёта

## 8. Сортировка данных

Произвести сортировку данных с Листа5 по столбцу "Абсолютная успеваемость, проц." (таблица 5, лист6).

Необходимо отсортировать данные, полученные в предыдущей программе по возрастанию. При сортировке двумерного массива ипользуется вложенные циклы. Во внешнем цикле используется оператор Do While…Lope, заключённые в нём операторы выполняются до тех пор, пока остаётся истинным условие While. Внутри цикла Do While выполняется два цикла For…Next.

Синтаксис:

*For Счётчик1=<начальное значении параметра цикла> To <конечное значение> [Step Шаг]*

*For Счётчик2=<начальное значении параметра цикла> To <конечное значение> [Step Шаг]*

*[Оператор (ы) тела цикла]*

*Next Счётчик2*

*Next Счётчик1*

Счётчик по параметру i-количество строк - сортируемый признак предыдущей строки сравнивается с признаком последующей строки. Если признаки первой строки больше признака второй строки, то выполняется цикл по j-количество столбцов, в котором данные этих строк меняются местами.

*Sub Сортировка ()*

*Sheets ("Лист6"). Select*

*Dim A () As Variant*

*n1 = Sheets ("Лист4"). Cells (5,12)*

*m = Sheets ("Лист2"). Cells (5,12)*

*ReDim A (1 To n1, 1 To m)*

*VVOD "Лист5", A, n1, m, 4*

*VIVOD "Лист6", A, n1, m, 4*

*ReDim A (1 To n1, 1 To m)*

*For i = 1 To n1*

*For j = 1 To m*

*A (i, j) = Sheets ("Лист5"). Cells (i + 4, j)*

*Sheets ("Лист6"). Cells (i + 4, j) = A (i, j)*

*Next j*

*Next i*

*Z = 1*

*Do While Z = 1*

*Z = 0*

*For i = 1 To n1 - 1*

*If Sheets ("Лист6"). Cells (i + 4,7) > Sheets ("Лист6"). Cells (i + 4 + 1,7) Then*

*For j = 1 To m*

*AZ = Sheets ("Лист6"). Cells (i + 4, j)*

*Sheets ("Лист6"). Cells (i + 4, j) = Sheets ("Лист6"). Cells (i + 4 + 1, j)*

*Sheets ("Лист6"). Cells (i + 4 + 1, j) = AZ*

*Next j*

*Z = 1*

*End If*

*Next i*

*Loop*

*End Sub*



Рис.6. Данные таблицы после сортировки

## 9. Создание автоматического макроса по сортировке

Создать автоматический макрос по сортировке по столбцу "Абсолютная успеваемость, проц." (Листа5, табл.6)

Встаем на лист, где будет макрос. Включаем запись макроса Сервис→Макрос→Начать запись→ОК. Появится квадрат, где кнопка остановить запись. На Листе5 (отчет) выделяем таблицу без заголовков и итогов, копируем на Лист9 (автосортировка) в элементе меню выбираем Данные→Сортировка→по возрастанию→по столбцам→ОК. Отмечаем столбец по которому будем сортировать. Нажимаем кнопку остановить запись.

*Sub Макрос1Сортировка ()*

*'*

*' Макрос1Сортировка Макрос*

*'*

*'*

*Sheets ("Лист5"). Select*

*Range ("A2: H17"). Select*

*Selection. Copy*

*Sheets ("Лист7"). Select*

*Range ("A2: A3"). Select*

*ActiveSheet. Paste*

*Application. CutCopyMode = False*

*Range ("A5: H17"). Select*

*ActiveWorkbook. Worksheets ("Лист7"). Sort. SortFields. Clear*

*ActiveWorkbook. Worksheets ("Лист7"). Sort. SortFields. Add Key: =Range ("A5"), \_*

*SortOn: =xlSortOnValues, Order: =xlAscending, DataOption: =xlSortNormal*

*With ActiveWorkbook. Worksheets ("Лист7"). Sort*

*. SetRange Range ("A4: H17")*

*. Header = xlYes*

*. MatchCase = True*

*. Orientation = xlTopToBottom*

*. SortMethod = xlPinYin*

*. Apply*

*End With*

*End Sub*



Рис.7. Данные таблицы после авто-сортировки

## 10. Выборка данных

Создать отчёт по выборке с Листа5 по столбцу "Качественная успеваемость, проц." (с Листа 8, табл.7)

Для того чтобы произвести выборку данных необходимо выполнить следующие действия:

Определить количество элементов нового массива по заданному условию введя переменную с помощью оператора InputBox

Объявить и переобъявить новый массив

Сформировать новый массив. Для этого необходимо задать номер первого элемента нового массива u=1. Затем выполняется цикл, в котором записывается условие выборки по столбцу "Качественная успеваемость, проц. ". Если результат проверки истина, то элемент анализируемого массива становится элементом нового массива.

Вывести новый элемент на Лист 8

*Sub ОтчётВыборка ()*

*Sheets ("Лист8"). Select*

*Dim A () As Variant*

*n1 = Sheets ("Лист4"). Cells (5,12)*

*m = Sheets ("Лист2"). Cells (5,12)*

*ReDim A (1 To n1, 1 To m)*

*VVOD "Лист5", A, n1, m, 4*

*C = InputBox ("Введите условие ")*

*Sheets ("Лист8"). Cells (5,11) = C*

*d = 0*

*For i = 1 To n1*

*If A (i,*

*8) > Sheets ("Лист8"). Cells (5,11) Then*

*d = d + 1*

*End If*

*Next i*

*Sheets ("Лист8"). Cells (5,10) = d*

*Dim B () As Variant*

*ReDim B (1 To d, 1 To m)*

*u = 1*

*For i = 1 To n1*

*If A (i,*

*8) > Sheets ("Лист8"). Cells (5,11) Then*

*For j = 1 To m*

*B (u, j) = A (i, j)*

*Next j*

*u = u + 1*

*End If*

*Next i*

*S = 0*

*For i = 1 To d*

*For j = 1 To m*

*Sheets ("Лист8"). Cells (i + 4, j) = B (i, j)*

*Next j*

*Next i*

*End Sub*

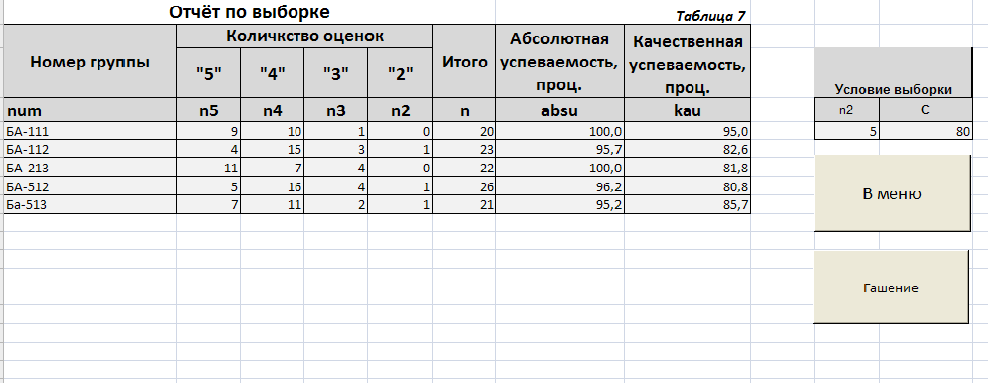


Рис.6. Данные таблицы после выборки

## 11. Создание автоматического макроса по выборке

Включаем запись макроса. Сервис→Макрос→Начать запись→ОК. Появится квадрат, где кнопка остановить запись. На Листе5 (отчет) выделяем таблицу без заголовков и итогов, копируем на Лист10 (автовыборка). Выделяем таблицу без заголовков и в элементе меню, выбираем Данные→Фильтр→Автофильтр→выбираем условие→ОК. Отмечаем столбец по которому будем сортировать. Завершаем работу макроса.

*Sub Макрос2Выборка ()*

*'*

*' Макрос2Выборка Макрос*

*'*

*'*

*Sheets ("Лист5"). Select*

*Selection. Copy*

*Sheets ("Лист9"). Select*

*ActiveSheet. Paste*

*Range ("H5: H17"). Select*

*Application. CutCopyMode = False*

*Selection. AutoFilter*

*ActiveSheet. Range ("$H$5: $H$17"). AutoFilter Field: =1, Criteria1: =">80", \_*

*Operator: =xlAnd*

*Range ("G22"). Select*

*End Sub*



Рис.7. Данные таблицы после авто-выборки

## 12. Определение максимального и минимального значения

Определить max и min значения по столбцам "Итого", "Абсолютная успеваемость, проц. ", "Качественная успеваемость" (таблица 9, лист 10)

Для определения max и min значений необходимо выполнить следующие действия:

Задать эталоновую переменную, которая будет текущем минимумом (максимумом)

Поочерёдно сравнивается каждый элемент совокупности с текущем минимумом (максимумом), и если этот элемент не удовлетворяет условиям поиска (в случаи минимума он больше, а в случае максимума - меньше), то происходит замена значения эталона на значение сравниваемого элемента

После полного просмотра всех элементов в переменной текущего минимума (максимума) находится действительный минимум (максимум)

Значение минимума (максимума) выводятся в соответствующие ячейки

*Sub minmax ()*

*Dim A () As Variant*

*n1 = Sheets ("Лист4"). Cells (5,12)*

*m = Sheets ("Лист2"). Cells (5,12)*

*ReDim A (1 To n1, 1 To m)*

*VVOD "Лист5", A, n1, m, 4*

*VIVOD "Лист10", A, n1, m, 4*

*VVOD "Лист10", A, n1, m, 4*

*For j = 3 To m*

*maxA = 0.00001*

*minA = 1000000*

*For i = 1 To n1*

*If A (i, j) > maxA Then*

*maxA = A (i, j)*

*End If*

*If A (i, j) < minA Then*

*minA = A (i, j)*

*End If*

*Next i*

*Sheets ("Лист10"). Cells (i + 4 + 2, j) = maxA*

*Sheets ("Лист10"). Cells (i + 4 + 3, j) = minA*

*Next j*

*End Sub*

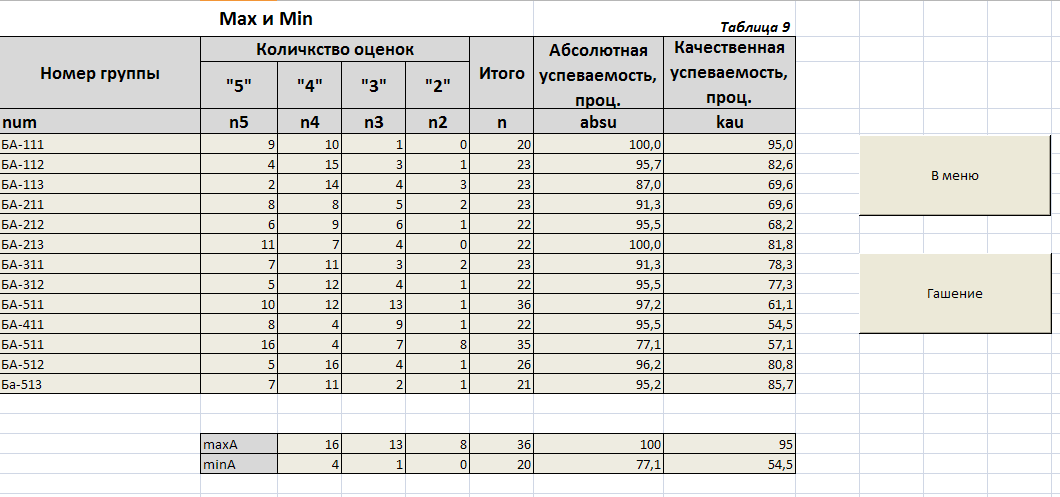


Рис.8. Данные таблицы после определения мин. и макс.

## 13. Создание диаграмм (гистограмма, круговая, смешанная)

Для создания диаграмм необходимо выбрать мастер диаграмм: Вставка-Диаграмма

или с помощь. Значка Диаграмма на панели инструментов.

Построение диаграммы состоит в выполнении четырёх шагов:

Выбор типа диаграмм

Задание диапазона

Записать параметры диаграммы заполнив строки заголовков

Размещение диаграммы

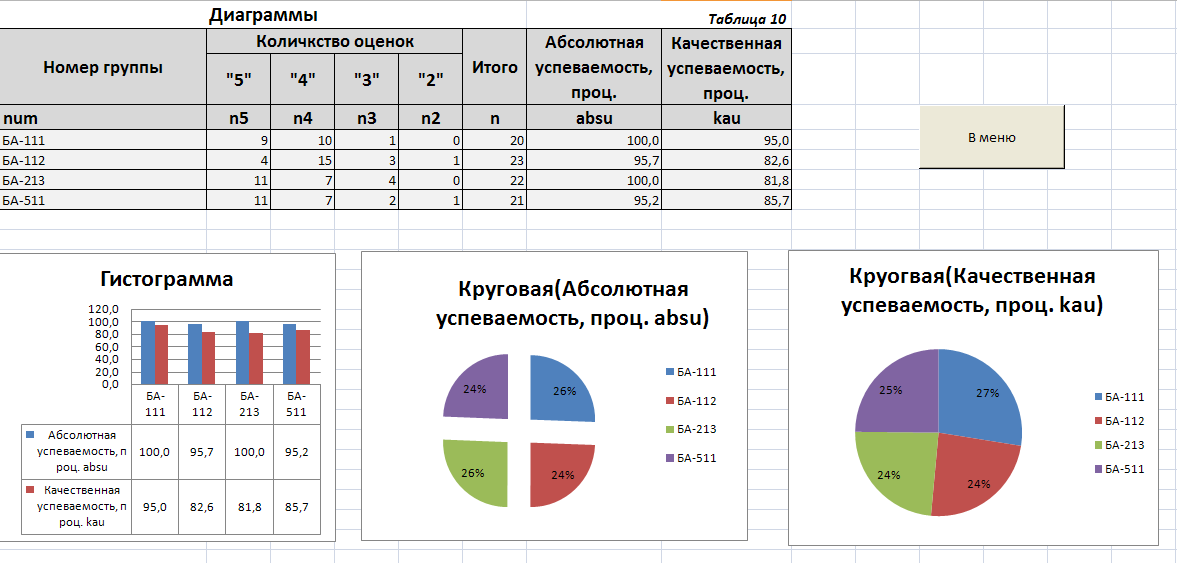


Рис.9. Диаграммы

## 14. Создание управляющих кнопок

Для удобства вызова и работы с программами целесообразно использовать управляющие кнопки для каждой программы.

Для создания кнопки необходимо выполнить следующее: элемент меню Вид → Панель инструментов → Формы. В появившейся панели форм выбирается значок, отвечающий за создание кнопки. Нажав на значок левой клавишей мыши, на листе вместо курсора появляется знак "+", который необходимо подвести на то место, где "+" вычерчивает прямоугольник, в нем фиксируется запись "кнопка 1". Аналогично создаются все кнопки управления. Для запуска любой программы не нужно заходить в Visual Basic, достаточно щелкнуть кнопку с ее названием.

После создания кнопок, кнопки можно отредактировать, переименовать, выбрать нужный шрифт, его размер. Для этого вызывается контекстное меню, в нем формат объекта.

Для того чтобы при нажатии кнопки происходила активизация программы, необходимо соединить (увязать) соответствующий макрос (программу) с кнопкой. При этом выполняются следующие действия:

щелкнуть правой клавишей мыши по кнопке для появления контекстного меню;

выбрать элемент меню "назначить макрос", появляется окно диалога под названием "назначить макрос объекту";

выбрать и подщелкнуть левой клавишей мыши нужное имя макроса, за активизацию которого и будет отвечать создаваемая кнопка;

ОК.

Для активизации листа в каждой программе нужно написать команду: *Sheets ("ЛистN"). Select*

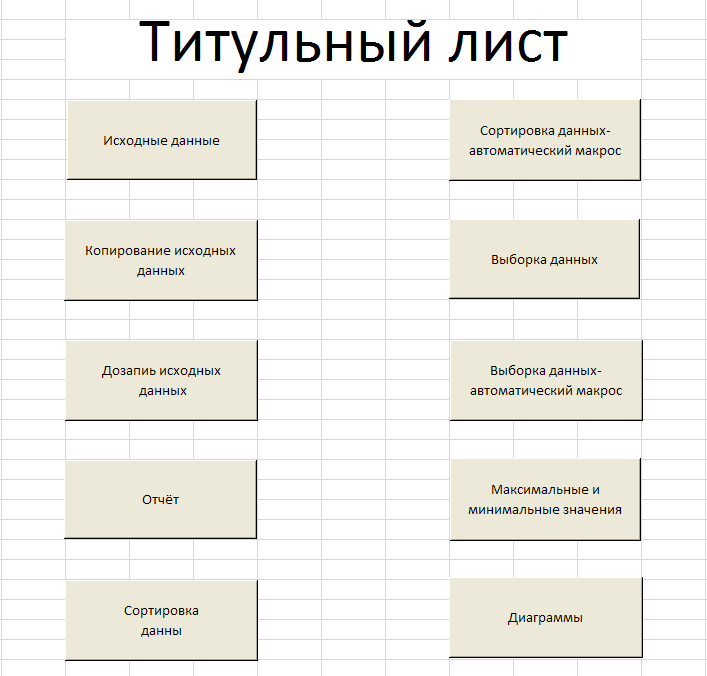


Рис.10. Экран управляющих кнопок

## 

## 15. Автоматический макрос Гашение

Для того чтобы произвести гашение данных необходимо выполнить следующие действия:

Включаем запись макроса

Выделяем диапазон гашения и нажимаем кнопку Delete.

Завершаем работу макроса.

Создаем кнопку "Гашение" и присваиваем ей соответствующий макрос.

## Заключение

В данной работе были рассмотрены основы программирования в среде VBA на примере решения экономических задач с использованием инструментария, макросов, операторов, функций и различных возможностей Exel

Таким образом, курсовая работа помогла приобрести опыт, касающийся работы в электронных таблицах Excel и базовых принципов программирования в среде VBA.

## Список использованной литературы

1. Александр Левин. Самоучитель работы на компьютере.7-е издание. -

М.: 2002

2. Хорев В.Д. Самоучитель программирования на VBA в Microsoft Office. - Киев: Юниор, 2001 г. - 320 с.

3. Уэллс Э., Харшбаргер С. Microsoft Excel 97. - М.: 1998.

4. Макарова Е.А., Швецова М.Н. Решение экономических задач и бухгалтерских задач с использованием инструментария VBA: метод. указ. и задания к выполнению курсовой работы, лабораторных работ по курсу "Информатика". - Новосибирск: Изд-во СГУПСа, 2008 г. - 47 с.

5. Конспекты лекций по информатике