**РГЗ по информатике**

**для студентов заочного отделения**

**(вариант для каждого задания по номеру в журнале)**

**Содержание работы**

**Цель работы**

Закрепление знаний и практических навыков работы на персональном компьютере с использованием языков программирования высокого уровня.

**Задания к контрольной работе**

Контрольная работа состоит из пяти заданий, решение которых выполняется с использованием языка программирования Visual Basic. К контрольной работе составляется пояснительная записка, которая должна быть подготовлена в MS Word.

|  |  |
| --- | --- |
| **Задание 1**. | Работа с двумерными массивами. |
| **Задание 2**. | Работы с процедурами и функциями. |
| **Задание 3**. | Работа с записями и файловыми типами данных. |
| **Задание 4**. | Работа с файлами и окнами |
| **Задание 5**. | Работа с графикой |

**Требования к выполнению и оформлению контрольной работы**

Работа должна быть представлена на бумажном и электронном носителях и представлять собой тексты программ на языке C++ и отчеты, содержание которых описано ниже.

**Требования к представлению и оформлению контрольной работы**

Работа должна быть представлена на бумажном (пояснительная записка) и электронном (в виде файлов) носителях. Пояснительная записка должна быть отпечатана на стандартных листах формата А4 (210×297 мм). На дискете должны быть файлы проектов решения задач в Visual Basic и исполняемые файлы (.exe), а также пояснительная записка, подготовленная в MS Word. При необходимости файлы могут быть заархивированы.

Текст пояснительной записки должен быть набран с учетом следующих параметров: поля документа сверху - 2,5 см, снизу - 2,5 см, слева - 3,0 см, справа - 1,5 см, шрифт Times New Roman, размер шрифта 14, межстрочный интервал 1, выравнивание - по ширине страницы, автоматическая нумерация страниц. Рисунки и таблицы должны иметь поясняющие надписи и быть пронумерованы.

Содержание пояснительной записки:

* титульный лист (см. образец);
* лист содержания (оглавление должно составляться с использованием средств автоматизации MS Word);
* основная часть, включающая пять разделов в каждом из которых приводится постановка задачи, описание ее решения, блок-схема, текст программы на Visual Basic и рисунки экранных форм разработанного проекта;
* список литературы (перечень использованных при выполнении контрольной работы материалов, на которые должны быть ссылки в тексте).

**Порядок защиты работы**

* К защите обязательно представить пояснительную записку и электронную версию работы.
* Работа защищается с использованием компьютера (студент демонстрирует полученные знания и практические навыки работы на компьютере, отвечая на вопросы преподавателя, принимающего контрольную работу).
* По результатам защиты выставляется зачет за контрольную работу.

**Задание 1. «Массивы»**

**Варианты**

1. Сколько положительных элементов содержит матрица размера 10×10, если *a*[*i*,*j*]=sin(*i*+*j*/2).
2. Определить сумму отрицательных элементов матрицы размера 20×10, если *a*[*i*,*j*]=cos(*i*-5×*j*).
3. В матрице размера 3×5 найти среднее арифметическое каждого из столбцов.
4. В матрице размера 4×3 найти значение максимального элемента в каждой строке.
5. В матрице размера 4×3 найти разность между суммой и произведением элементом.
6. Найти сумму и количество положительных элементов матрицы размера 4×4.
7. Найти место расположения максимального по модулю элемента матрицы размера 5×3 указав номер строки и столбца.
8. Найти разность между максимальным элементом матрицы размера 4×4 и суммой ее положительных элементов.
9. Определить сумму и произведение элементов матрицы размера 4×5, имеющих оба четных индекса.
10. Найти сумму элементов матрицы размера 4×4 в четных строках и произведение элементов в нечетных строках.
11. Найти среднее арифметическое наибольшего и наименьшего значений элементов матрицы размера 4×3.
12. В матрице размера 4×5 найти сумму наибольших значений элементов ее столбцов.
13. В матрице размера 6×3 найти номера строк, все элементы которых положительны.
14. В матрице размера 4×4 найти номера (*i*,*j*) и определить сумму отрицательных элементов.
15. Преобразовать матрицу размера 4×4 по правилу: вторую строку сделать четвертым столбцом, а четвертый столбец - второй строкой.
16. В матрице размера 4×4 найти сумму элементов над главной диагональю.
17. В матрице размера 4×4 найти максимальный по модулю элемент на побочной диагонали.
18. Преобразовать матрицу размера 4×4, поменяв местами в каждой строке элементы главной и побочной диагонали.
19. Определить является ли целочисленная матрица размером 4×4 симметричной относительно главной диагонали.
20. Сделать целочисленную матрицу размером 4×4 симметричной относительно главной диагонали, заменяя при отсутствии симметрии меньший элемент на больший.
21. В целочисленной матрице размера 4×5 все элементы с наибольшим значением заменить суммой положительных элементов.
22. В целочисленной матрице размера 4×5 определить количество наибольших по модулю элементов и их номера.
23. В матрице размером 5×3 поменять местами строку с наибольшим элементом и строку с наименьшим элементом.
24. В матрице размера 4×3, все элементы которые различны, найти произведение элементов строки, содержащие наибольший по модулю элемент матрицы.
25. Если в матрице размера 4×4 произведение элементов побочной диагонали отрицательно, то все элементы на главной диагонали заменить на 0, если положительно - на 1.
26. В символьной матрице размера 5×5 заменить все буквы *W* на букву *Q*, если количество букв *W* больше 3.
27. В символьной матрице размера 4×4 определить количество цифр и если оно больше 2, то все цифры заменить на &.
28. В символьной матрице размера 5×5 определить количество цифр и все цифры заменить латинскими буквами по правилу: 1-*А*; 2-*В*; 3-*С*... 0-*J* (не используя 10 раз подряд оператор IF).
29. Определить, сколько прописных букв латинского алфавита содержит символьная матрица 4×5.
30. Определить сумму и произведение цифр, расположенных на главной диагонали символьной матрицы размера 5×5.

**Задание 2. «Процедуры и функции»**

**Варианты**

1. Определить подпрограмму вычисления площади правильного *N*-угольника, вписанного в окружность заданного радиуса.
2. Определить подпрограмму вычисления площади кольца при известном наружном и внутреннем диаметре.
3. Определить подпрограмму вычисления площади сектора при известном наружном, внутреннем диаметре и центральном угле.
4. Определить подпрограмму вычисления расстояния между двумя точками в пространстве с координатами (*x*1,*y*1,*z*1) и (*x*2,*y*2,*z*2).
5. Дано целое число *N* (*N*<100), определяющее возраст человека (в годах). Определить подпрограмму, дающую для этого числа наименования «год», «года» или «лет»: например 1 год, 23 года, 45 лет и т.д.
6. Определить подпрограмму вычисления факториала заданного целочисленного выражения, например, вычислить (2×*K*+1)!.
7. Определить подпрограмму вычисления корней квадратного уравнения *ax*2+*bx*+*c*=0.
8. Даны вещественные числа *X*1, *Y*1, *X*2, *Y*2, ..., *X*8, *Y*8. Найти периметр восьмиугольника, вершины которого имеют координаты (*X*1,*Y*1), (*X*2,*Y*2), ..., (*X*8,*Y*8). (Определить подпрограмму вычисления расстояния между двумя точками, заданными своими координатами.)
9. Даны целое число *N* и вещественные числа *X*1, *Y*1, *X*2, *Y*2, ..., *XN*, *YN*. Найти площадь *N*-угольника, вершины которого имеют соответственно координаты (*X*1,*Y*1), (*X*2,*Y*2), ..., (*XN*,*YN*). (Определить подпрограмму вычисления площади треугольника по заданными координатам вершин.)
10. Дано целое число *N*. Выяснить, имеются ли среди чисел *N*, *N*+1, ..., 2*N* близнецы, т.е. простые числа, разность между которыми равна двум. (Определить процедуру, позволяющую распознавать простые числа.)
11. Составить процедуру вычисления значения целого числа по заданной строке символов, являющихся записью этого числа:
    * в десятичной системе счисления;
    * в шестнадцатеричной системе счисления (шестнадцатеричные цифры - это цифры от 0 до 9 и буквы от *A* до *F*).
12. Определить функцию, которая подсчитывает количество литер в текстовом файле (исключая цифры).
13. Определить процедуру замены в текстовом файле одного заданного символа на другой.
14. Определить процедуру, подсчитывающую количество цифр в каждой строке текстового файла.
15. Написать процедуры шифровки и дешифровки текстового файла.
16. Определить подпрограмму вычисления площади правильного *N*-угольника, вписанного в окружность заданного радиуса.
17. Определить подпрограмму вычисления площади кольца при известном наружном и внутреннем диаметре.
18. Определить подпрограмму вычисления площади сектора при известном наружном, внутреннем диаметре и центральном угле.
19. Определить подпрограмму вычисления расстояния между двумя точками в пространстве с координатами (*x*1,*y*1,*z*1) и (*x*2,*y*2,*z*2).
20. Дано целое число *N* (*N*<100), определяющее возраст человека (в годах). Определить подпрограмму, дающую для этого числа наименования «год», «года» или «лет»: например 1 год, 23 года, 45 лет и т.д.
21. Определить подпрограмму вычисления факториала заданного цело численного выражения, например вычислить (2×*K*+1)!.
22. Определить подпрограмму вычисления корней квадратного уравнения *ax*2+*bx*+*c*=0.
23. Даны вещественные числа *X*1, *Y*1, *X*2, *Y*2, ..., *X*8, *Y*8. Найти периметр восьмиугольника, вершины которого имеют координаты (*X*1,*Y*1), (*X*2,*Y*2), ..., (*X*8,*Y*8). (Определить подпрограмму вычисления расстояния между двумя точками, заданными своими координатами.)
24. Даны целое число *N* и вещественные числа *X*1, *Y*1, *X*2, *Y*2, ..., *XN*, *YN*. Найти площадь *N*-угольника, вершины которого имеют соответственно координаты (*X*1,*Y*1), (*X*2,*Y*2), ..., (*XN*,*YN*). (Определить подпрограмму вычисления площади треугольника по заданными координатам вершин.)
25. Дано целое число *N*. Выяснить, имеются ли среди чисел *N*, *N*+1, ..., 2*N* близнецы, т.е. простые числа, разность между которыми равна двум. (Определить процедуру, позволяющую распознавать простые числа.)
26. Составить процедуру вычисления значения целого числа по заданной строке символов, являющихся записью этого числа:
    * в десятичной системе счисления;
    * в шестнадцатеричной системе счисления (шестнадцатеричные цифры - это цифры от 0 до 9 и буквы от *A* до *F*).
27. Определить функцию, которая подсчитывает количество литер в текстовом файле (исключая пробелы).
28. Определить процедуру замены в текстовом файле одного заданного символа на другой.
29. Определить процедуру, подсчитывающую количество цифр в каждой строке текстового файла.
30. Написать процедуры шифровки и дешифровки текстового файла.

**Задание 3. «Записи и файлы»**

**Варианты**

1. **Абитуриенты.**  
   ***Запись***: Ф.И.О., год рождения, год окончания школы, средний балл по аттестату.  
   ***Результаты обработки данных:*** список со средним баллом >4,5 и количество отличников.
2. **Абитуриенты.**  
   ***Запись***: Ф.И.О., год рождения, какое учебное заведение окончил, место проживания (город, село).  
   ***Результаты обработки данных***: список абитуриентов, проживающих не в Уфе и количество окончивших техникум.
3. **Абитуриенты.**  
   ***Запись***: Ф.И.О., год окончания школы, средний балл по аттестату, служба в армии.  
   ***Результаты обработки данных***: список окончивших школу до 1990 года и количество служивших в армии.
4. **Абитуриенты.**  
   ***Запись***: Ф.И.О., год рождения, пол, средний балл по аттестату.  
   ***Результаты обработки данных:***список абитуриентов моложе 18 лет и количество абитуриентов мужского пола.
5. **Абитуриенты.**  
   ***Запись***: Ф.И.О., год рождения, средний балл по аттестату, баллы на вступительных экзаменах (4 экзамена).  
   ***Результаты обработки данных:***список абитуриентов набравших проходной балл (18) и количество сдавших экзамены без троек.
6. **Абитуриенты.**  
   ***Запись***: Ф.И.О., пол, средний балл по аттестату, баллы на вступительных экзаменах (4 экзамена).  
   ***Результаты обработки данных***: список абитуриентов со средним баллом по аттестату >4,75 и количество сдавших экзамены только на «отлично».
7. **Студенты.**  
   ***Запись***: Ф.И.О., пол, место проживания (дома, в общежитии, на квартире), получает ли стипендию.  
   ***Результаты обработки данных***: список студентов проживающих в общежитии и количество получающих стипендию.
8. **Студенты.**  
   ***Запись***: Ф.И.О., год рождения, пол, место проживания (дома, в общежитии, на квартире).  
   ***Результаты обработки данных***: список студентов моложе 18 лет и процент проживающих в общежитии.
9. **Студенты.**  
   ***Запись***: Ф.И.О., пол, год рождения, баллы на экзаменах (4 экзамена).  
   ***Результаты обработки данных***: список студентов получивших на экзаменах 4 и 5 и процент сдавших без троек.
10. **Студенты.**  
    ***Запись***: Ф.И.О., год рождения, место проживания (дома, в общежитии, на квартире), баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов получивших на экзаменах 2 и процент сдавших с первого предъявления.
11. **Студенты.**  
    ***Запись***: Ф.И.О., пол, баллы на экзаменах (4 экзамена), получает ли стипендию.  
    ***Результаты обработки данных***: список студентов получающих стипендию и процент сдавших экзамены на «отлично».
12. **Студенты.**  
    ***Запись***: Ф.И.О., пол, год рождения, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов мужского пола и процент сдавших экзамены на 4 и 5.
13. **Студенты.**  
    ***Запись***: Ф.И.О., место проживания (дома, в общежитии, на квартире), год рождения, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов проживающих в общежитии и процент получивших на экзамене 2.
14. **Студенты.**  
    ***Запись***: Ф.И.О., место проживания (дома, в общежитии, на квартире), баллы на экзаменах (4 экзамена), получает ли стипендию.  
    ***Результаты обработки данных***: список студентов проживающих в общежитии и получающих стипендию процент сдавших экзамены на отлично.
15. **Студенты.**  
    ***Запись***: Ф.И.О., место проживания (дома, в общежитии, на квартире), пол, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов мужского пола проживающих в общежитии процент сдавших экзамены без троек.
16. **Абитуриенты.**  
    ***Запись***: Ф.И.О., год рождения, год окончания школы, средний балл по аттестату.  
    ***Результаты обработки данных:*** список со средним баллом >4,5 и количество отличников.
17. **Абитуриенты.**  
    ***Запись***: Ф.И.О., год рождения, какое учебное заведение окончил, место проживания (город, село).  
    ***Результаты обработки данных***: список абитуриентов, проживающих не в Уфе и количество окончивших техникум.
18. **Абитуриенты.**  
    ***Запись***: Ф.И.О., год окончания школы, средний балл по аттестату, служба в армии.  
    ***Результаты обработки данных***: список окончивших школу до 1990 года и количество служивших в армии.
19. **Абитуриенты.**  
    ***Запись***: Ф.И.О., год рождения, пол, средний балл по аттестату.  
    ***Результаты обработки данных:***список абитуриентов моложе 18 лет и количество абитуриентов мужского пола.
20. **Абитуриенты.**  
    ***Запись***: Ф.И.О., год рождения, средний балл по аттестату, баллы на вступительных экзаменах (4 экзамена).  
    ***Результаты обработки данных:***список абитуриентов набравших проходной балл (18) и количество сдавших экзамены без троек.
21. **Абитуриенты.**  
    ***Запись***: Ф.И.О., пол, средний балл по аттестату, баллы на вступительных экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список абитуриентов со средним баллом по аттестату >4,75 и количество сдавших экзамены только на «отлично».
22. **Студенты.**  
    ***Запись***: Ф.И.О., пол, место проживания (дома, в общежитии, на квартире), получает ли стипендию.  
    ***Результаты обработки данных***: список студентов проживающих в общежитии и количество получающих стипендию.
23. **Студенты.**  
    ***Запись***: Ф.И.О., год рождения, пол, место проживания (дома, в общежитии, на квартире).  
    ***Результаты обработки данных***: список студентов моложе 18 лет и процент проживающих в общежитии.
24. **Студенты.**  
    ***Запись***: Ф.И.О., пол, год рождения, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов получивших на экзаменах 4 и 5 и процент сдавших без троек.
25. **Студенты.**  
    ***Запись***: Ф.И.О., год рождения, место проживания (дома, в общежитии, на квартире), баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов получивших на экзаменах 2 и процент сдавших с первого предъявления.
26. **Студенты.**  
    ***Запись***: Ф.И.О., пол, баллы на экзаменах (4 экзамена), получает ли стипендию.  
    ***Результаты обработки данных***: список студентов получающих стипендию и процент сдавших экзамены на «отлично».
27. **Студенты.**  
    ***Запись***: Ф.И.О., пол, год рождения, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов старше 18 лет и процент сдавших экзамены на 4 и 5.
28. **Студенты.**  
    ***Запись***: Ф.И.О., место проживания (дома, в общежитии, на квартире), год рождения, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов проживающих в общежитии и процент получивших на экзамене 2.
29. **Студенты.**  
    ***Запись***: Ф.И.О., место проживания (дома, в общежитии, на квартире), баллы на экзаменах (4 экзамена), получает ли стипендию.  
    ***Результаты обработки данных***: список студентов проживающих в общежитии и получающих стипендию процент сдавших экзамены на отлично.
30. **Студенты.**  
    ***Запись***: Ф.И.О., место проживания (дома, в общежитии, на квартире), пол, баллы на экзаменах (4 экзамена).  
    ***Результаты обработки данных***: список студентов мужского пола проживающих в общежитии процент сдавших экзамены без троек.

**Задание 4. «Текстовые файлы и окна»**

**Варианты**

1. Написать процедуру для ввода из файла квадратной матрицы размерности *N* (*N*<50). Вывести матрицу в окно на экране.
2. Написать процедуру для ввода из файла прямоугольной матрицы размерности *N*×*M* (*N*<50, *M*<50). Вывести матрицу в окно на экране.
3. Написать процедуру для ввода из окна на экране квадратной матрицы размерности *N* (*N*<50). Записать матрицу в текстовый файл.
4. Написать процедуру для ввода из окна на экране прямоугольной матрицы размерности *N*×*M* (*N*<50, *M*<30). Записать матрицу в текстовый файл, и вывести в другое окно на экране дисплея.
5. Написать процедуру для ввода из файла квадратной матрицы размерности *N* (*N*<50). Написать функцию вычисления суммы элементов матрицы на главной диагонали. Вывести матрицу в одно окно на экране, а вычисленную сумму - в другое.
6. Написать процедуру для ввода из файла прямоугольной матрицы размерности *N*×*M* (*N*<40, *M*<30). Написать функцию вычисления максимального элемента матрицы. Вывести матрицу в одно окно на экране, а значение максимального элемента - в другое.
7. Написать процедуру для ввода из файла прямоугольной матрицы размерности *N*×*M* (*N*<40, *M*<50). Написать функцию вычисления количества максимальных элементов матрицы. Вывести матрицу в одно окно на экране, а значение количества максимальных элементов - в другое.
8. Написать процедуру для ввода из файла квадратной символьной матрицы размерности *N*×*N* (*N*<30). Написать функцию подсчета количества букв латинского алфавита. Вывести матрицу в одно окно на экране, а значение количества букв латинского алфавита - в другое.
9. Написать процедуру для ввода из файла прямоугольной символьной матрицы размерности *N*×*M* (*N*<30, *M*<40). Написать функцию, заменяющую цифры на соответствующие буквы латинского алфавита. Вывести исходную матрицу в одно окно на экране, а полученную - в другое.
10. Написать процедуру для ввода из файла прямоугольной символьной матрицы размерности *N*×*M* (*N*<20, *M*<40). Написать функцию, вычисляющую сумму цифр, встречающихся в матрице. Вывести исходную матрицу в одно окно на экране, а вычисленную сумму - в другое.
11. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна - 76).
12. Написать процедуру для чтения текста из файла. Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 12, ширина окна должна определяться автоматически в зависимости от максимальной длины текстовой строки в файле).
13. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна должна определяться автоматически каждый раз в зависимости от максимальной длины текстовой строки).
14. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна 76). Предусмотреть перелистывание текста вперед и назад.
15. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне определяется по спец. символу в текстовом файле, ширина окна 76). Предусмотреть перелистывание текста вперед и назад.
16. Написать процедуру для ввода из файла квадратной матрицы размерности *N* (*N*<50). Вывести матрицу в окно на экране.
17. Написать процедуру для ввода из файла прямоугольной матрицы размерности *N*×*M* (*N*<50, *M*<50). Вывести матрицу в окно на экране.
18. Написать процедуру для ввода из окна на экране квадратной матрицы размерности *N* (*N*<50). Записать матрицу в текстовый файл.
19. Написать процедуру для ввода из окна на экране прямоугольной матрицы размерности *N*×*M* (*N*<50, *M*<30). Записать матрицу в текстовый файл, и вывести в другое окно на экране дисплея.
20. Написать процедуру для ввода из файла квадратной матрицы размерности *N* (*N*<50). Написать функцию вычисления суммы элементов матрицы на главной диагонали. Вывести матрицу в одно окно на экране, а вычисленную сумму - в другое.
21. Написать процедуру для ввода из файла прямоугольной матрицы размерности *N*×*M* (*N*<40, *M*<30). Написать функцию вычисления максимального элемента матрицы. Вывести матрицу в одно окно на экране, а значение максимального элемента - в другое.
22. Написать процедуру для ввода из файла прямоугольной матрицы размерности *N*×*M* (*N*<40, *M*<50). Написать функцию вычисления количества максимальных элементов матрицы. Вывести матрицу в одно окно на экране, а значение количества максимальных элементов - в другое.
23. Написать процедуру для ввода из файла квадратной символьной матрицы размерности *N*×*N* (*N*<30). Написать функцию подсчета количества букв латинского алфавита. Вывести матрицу в одно окно на экране, а значение количества букв латинского алфавита - в другое.
24. Написать процедуру для ввода из файла прямоугольной символьной матрицы размерности *N*×*M* (*N*<30, *M*<40). Написать функцию, заменяющую цифры на соответствующие буквы латинского алфавита. Вывести исходную матрицу в одно окно на экране, а полученную - в другое.
25. Написать процедуру для ввода из файла прямоугольной символьной матрицы размерности *N*×*M* (*N*<20, *M*<40). Написать функцию, вычисляющую сумму цифр, встречающихся в матрице. Вывести исходную матрицу в одно окно на экране, а вычисленную сумму - в другое.
26. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна - 76).
27. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна должна определяться автоматически в зависимости от максимальной длины текстовой строки в файле).
28. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна должна определяться автоматически каждый раз в зависимости от максимальной длины текстовой строки).
29. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна 76). Предусмотреть перелистывание текста вперед и назад.
30. Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне определяется по спец. символу в текстовом файле, ширина окна 76). Предусмотреть перелистывание текста вперед и назад.

**Задание 5. «Графика»**

**Варианты**

1. Построить график функции *Y*=3*x*2 на отрезке [-5;5].
2. Построить график функции *Y*=*x*+3х−2 на отрезке [3;10].
3. Построить график функции *Y*=3+*x*2+3*x*2 на отрезке [-3;-1].
4. Построить график функции *Y*=1*x*2−2*x*+1 на отрезке [-10;0].
5. Построить окружность радиуса *R*=30 c центром в начале координат по заданному параметрическому представлению *X*=*R*cos*t*; *Y*=*R*sin*t*, где 0≤*t*≤2π.
6. Построить эллипс с большой и малой полуосями, равными *R*1=10 и *R*2=30 и расположенными параллельно осям координат по заданному параметрическому представлению *X*=*R*1cos*t*; *Y*=*R*2sin*t*, где 0≤*t*≤2π.
7. Построить кардиоиду по заданному параметрическому представлению *X*=*a*cos*t*(1+cos*t*); *Y*=*a*sin*t*(1+cos*t*), где *a*=15; 0≤*t*≤2π.
8. Построить эпициклоиду по заданному параметрическому представлению *X*=(*a*+*b*)cos*t*−*a*cos*a*(*a*+*b*)*t* ; *Y*=(*a*+*b*)sin*t*−*a*sin*a*(*a*+*b*)*t* , где *a*=12; *b*=5; 0≤*t*≤2π.
9. Построить лемнискату по уравнениям в полярных координатах (*p* - радиус вектор; *f* - угол) *p*=*a*√2cos2*f* ; при *a*=5.
10. Построить спираль вокруг начала координат с *N* витками и внешним радиусом *R*. Начальное направление спирали образует с осью *X* угол 0 градусов. Параметрическое представление спирали *X*=*R*cos*t*; *Y*=*R*sin*t*;*R*=*t*2; 0≤*t*≤2*N*π.
11. Построить график функции *Y*=3*x*2 на отрезке [-5;5].
12. Построить график функции *Y*=*x*−3*x*+4 на отрезке [4;12].
13. Построить график функции *Y*=3+*x*2+3*x*2 на отрезке [-3;-1].
14. Построить график функции *Y*=1*x*2−2*x*+1 на отрезке [-10;0].
15. Построить окружность радиуса *R*=30 c центром в начале координат по заданному параметрическому представлению *X*=*R*cos*t*; *Y*=*R*sin*t*, где 0≤*t*≤2π.
16. Построить эллипс с большой и малой полуосями, равными *R*1=10 и *R*2=30 и расположенными параллельно осям координат по заданному параметрическому представлению *X*=*R*1cos*t*; *Y*=*R*2sin*t*, где 0≤*t*≤2π.
17. Построить кардиоиду по заданному параметрическому представлению *X*=*a*cos*t*(1+cos*t*); *Y*=*a*sin*t*(1+cos*t*), где *a*=15; 0≤*t*≤2π.
18. Построить эпициклоиду по заданному параметрическому представлению *X*=(*a*+*b*)cos*t*−*a*cos*a*(*a*+*b*)*t* ; *Y*=(*a*+*b*)sin*t*−*a*sin*a*(*a*+*b*)*t* , где *a*=12; *b*=5; 0≤*t*≤2π.
19. Построить лемнискату по уравнениям в полярных координатах (*p* - радиус вектор; *f* - угол) *p*=*a*√2cos2*f* ; при *a*=5.
20. Построить спираль вокруг начала координат с *N* витками и внешним радиусом *R*. Начальное направление спирали образует с осью *X* угол 0 градусов. Параметрическое представление спирали *X*=*R*cos*t*; *Y*=*R*sin*t*;*R*=*t*2; 0≤*t*≤2*N*π.
21. Построить график функции *Y*=3*x*2 на отрезке [-5;5].
22. Построить график функции *Y*=*x*+3х−2 на отрезке [3;10].
23. Построить график функции *Y*=3+*x*2+3*x*2 на отрезке [-3;-1].
24. Построить график функции *Y*=1*x*2−2*x*+1 на отрезке [-10;0].
25. Построить окружность радиуса *R*=30 c центром в начале координат по заданному параметрическому представлению *X*=*R*cos*t*; *Y*=*R*sin*t*, где 0≤*t*≤2π.
26. Построить эллипс с большой и малой полуосями, равными *R*1=10 и *R*2=30 и расположенными параллельно осям координат по заданному параметрическому представлению *X*=*R*1cos*t*; *Y*=*R*2sin*t*, где 0≤*t*≤2π.
27. Построить кардиоиду по заданному параметрическому представлению *X*=*a*cos*t*(1+cos*t*); *Y*=*a*sin*t*(1+cos*t*), где *a*=15; 0≤*t*≤2π.
28. Построить эпициклоиду по заданному параметрическому представлению *X*=(*a*+*b*)cos*t*−*a*cos*a*(*a*+*b*)*t* ; *Y*=(*a*+*b*)sin*t*−*a*sin*a*(*a*+*b*)*t* , где *a*=12; *b*=5; 0≤*t*≤2π.
29. Построить лемнискату по уравнениям в полярных координатах (*p* - радиус вектор; *f* - угол) *p*=*a*√2cos2*f* ; при *a*=5.
30. Построить спираль вокруг начала координат с *N* витками и внешним радиусом *R*. Начальное направление спирали образует с осью *X* угол 0 градусов. Параметрическое представление спирали *X*=*R*cos*t*; *Y*=*R*sin*t*;*R*=*t*2; 0≤*t*≤2*N*π.

**Пример выполнения задания 1 «Массивы»**

**1.1. Постановка задачи**

В матрице размера 5×4 найти сумму наибольших значений элементов ее строк.

**1.2. Блок-схема программы**

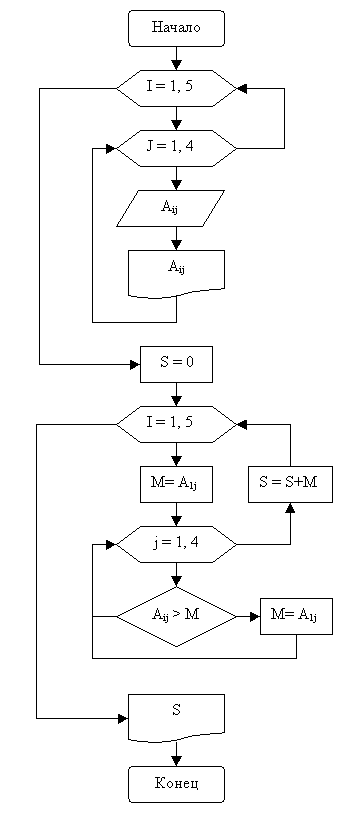


Рис.1. Блок-схема

**1.3. Исходные данные для отладки и тестирования программы**

В качестве исходных данных выберем матрицу с произвольными значениями и проверим работу программы. Эти данные можно посмотреть в рабочем окне программы на рис.2.

**1.4. Исходный текст программы**

Dim Am(5, 4), s As Double

Dim av() As String

Private Sub Command1\_Click()

Picture1.Cls

av = Split(Text1, "/") 'разбивает строку по разделителю и результат записывает в одномерный массив

k = 0

For i = 0 To 5 - 1

StrAm = ""

For j = 0 To 4 - 1

Am(i, j) = Val(av(k))

StrAm = StrAm + " " + Str(Am(i, j))

k = k + 1

Next

Picture1.Print StrAm

Next

End Sub

Private Sub Command2\_Click()

s = 0

For i = 0 To 5 - 1

MaxAm = Am(i, 1)

For j = 0 To 4 - 1

If Am(i, j) > MaxAm Then MaxAm = Am(i, j)

Next

s = s + MaxAm

Next

Picture2.Cls

Picture2.Print "Сумма = "; s

End Sub

Private Sub Command3\_Click()

End

End Sub

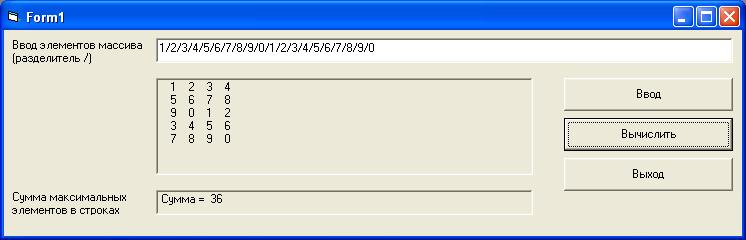


Рис.2. Экранная форма программы

**Пример выполнения задания 2 «Процедуры и функции»**

**2.1. Постановка задачи**

Определить функцию, которая подсчитывает количество литер в текстовом файле (исключая пробелы).

**2.2. Блок-схема программы**

В примере не приведена.

**2.3. Исходные данные для отладки**

Кнопка «Прочитать» открывает выбранный файл, выводит его содержимое в текстовое поле и позволяет выполнить алгоритм подсчета литер в текстовом файле.

**2.4. Исходный текст программы**

Dim StrTemp As String

Private Sub Command1\_Click()

Picture1.Cls

Open File1.Path + "\" + File1.FileName For Input As #1

S = 0

Text1.Text = ""

Do While Not(EOF(1))

Line Input #1, StrTemp

For i = 1 To Len(StrTemp)

z = Mid(StrTemp, i, 1)

If z <> "" Then S = S + 1

Next i

Text1.Text = Text1.Text + StrTemp + Chr$(13) + Chr$(10)

Loop

Close #1

Picture1.Print "Число литер, исключая пробелы равно "; S

End Sub

Private Sub Drive1\_Change()

Dir1.Path = Drive1.Drive ' Выбираем диск

End Sub

Private Sub Dir1\_Change()

File1.Path = Dir1.Path ' Выбираем папку

End Sub

Private Sub Command2\_Click()

End

End Sub

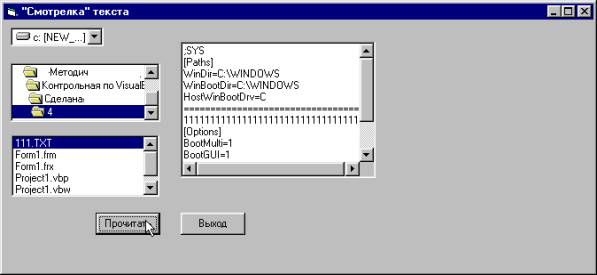


Рис.3. Результат выполнения программы

**Пример выполнения задания 3 «Записи и файлы»**

**3.1. Постановка задачи**

**Студенты**. ***Запись***: Ф.И.О., пол, год рождения, баллы на экзаменах (4 экзамена).

***Результаты обработки данных***: список студентов старше 18 лет и процент сдавших экзамены на 4 и 5.

**3.2. Блок-схема программы**

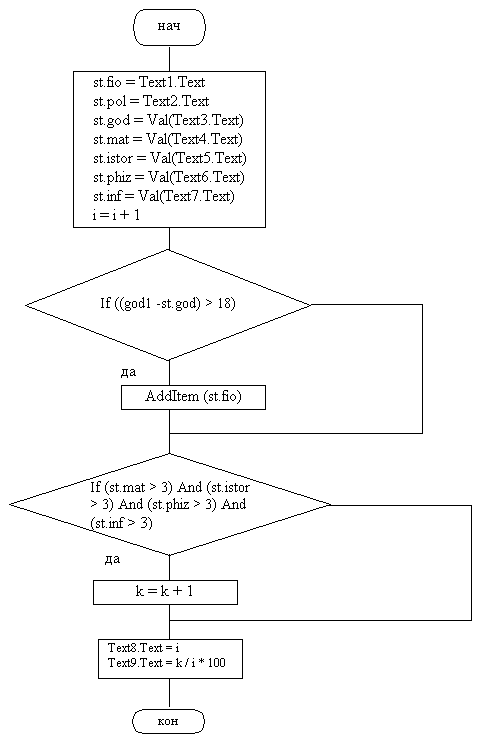


Рис.4. Блок-схема

**3.3. Исходный текст программы**

Dim st As stud

Dim i, k As Integer

Private Sub Command1\_Click

god1 = 2004

st.fio = Text1.Text

st.pol = Text2.Text

st.god = Val(Text3.Text)

st.mat = Val(Text4.Text)

st.istor = Val(Text5.Text)

st.phiz = Val(Text6.Text)

st.inf = Val(Text7.Text)

i = i + 1

If ((god1 - st.god) > 18) Then List1.AddItem(st.fio)

If (st.mat > 3) And (st.istor > 3) And (st.phiz > 3) And (st.inf > 3) Then k = k + 1

Text1.Text = ""

Text2.Text = "м"

Text3.Text = "1980"

Text4.Text = "5"

Text5.Text = "5"

Text6.Text = "5"

Text7.Text = "5"

End Sub

Private Sub Command2\_Click()

Text8.Text = i

Text9.Text = k / i \* 100

End Sub

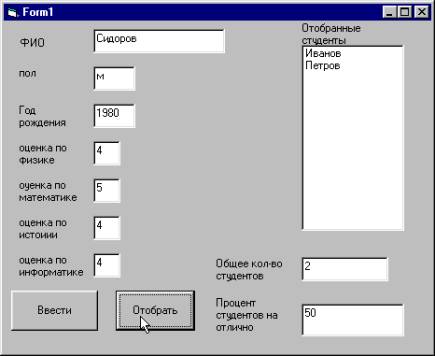


Рис.5. Результат выполнения программы

**Пример выполнения задания 4 «Текстовые файла и окна»**

**4.1. Постановка задачи**

Написать процедуру для чтения текста из файла (количество строк не более 100). Написать процедуру, выводящую текст в окно на экране дисплея (число строк в окне 10, ширина окна должна определяться автоматически в зависимости от максимальной длины текстовой строки в файле).

**4.2. Блок-схема программы**

В примере не приведена.

**4.3. Исходный текст программы**

Dim StrTemp As String

Private Sub Command1\_Click()

file = File1.Path + "\" + File1.FileName

Open file For Input As #1

Do While Not (EOF(1))

Line Input #1, StrTemp

If maxSTR < Len(StrTemp) Then maxSTR = Len(StrTemp)

Loop

Close #1

Text1.Width = maxSTR \* 100

Open file For Input As #2

Do While Not (EOF(2))

Line Input #2, StrTemp

Text1.Text = Text1.Text + StrTemp + Chr$(13) + Chr$(10)

Loop

Close #2

End Sub

Private Sub Drive1\_Change()

Dir1.Path = Drive1.Drive ' Выбираем диск

End Sub

Private Sub Dir1\_Change()

File1.Path = Dir1.Path ' Выбираем папку

End Sub

Private Sub Command2\_Click()

End

End Sub



Рис.6. Настройка свойств текстового поля

Для размещения на форме элементов интерфейса программы. Позволяющих выполнять визуальный поиск файла на дисках и в каталогах ПК использовано окно выбора компонентов (рис.7).

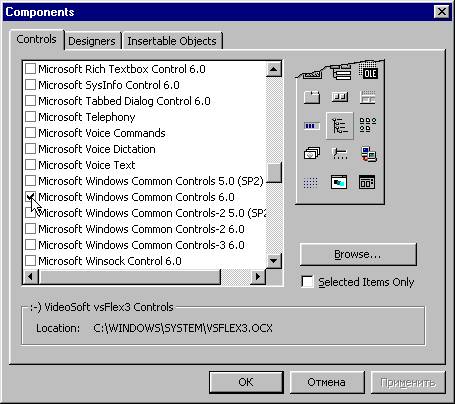


Рис.7. Окно выбора компонентов интерфейса

Для поиска файла выбирается диск, затем папка на диске и в ней - текстовый файл. Кнопка «Прочитать» позволяет вывести содержимое текстового файла в окно программы, ширина которого меняется в зависимости от максимальной длины строки текста в файле.

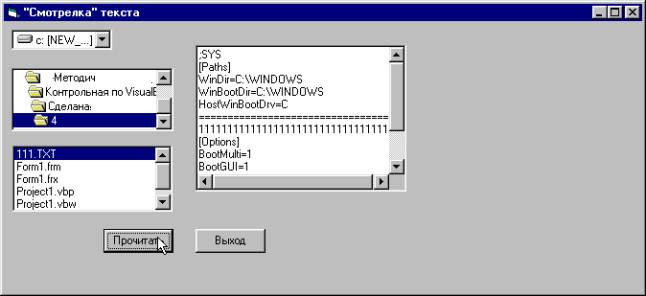


Рис.8. Результат выполнения программы

**Пример выполнения задания 5 «Графика»**

**5.1. Постановка задачи**

Построить график функции *Y*=*x*−2*x*+3 на отрезке [3;10].

**5.2. Блок-схема программы**

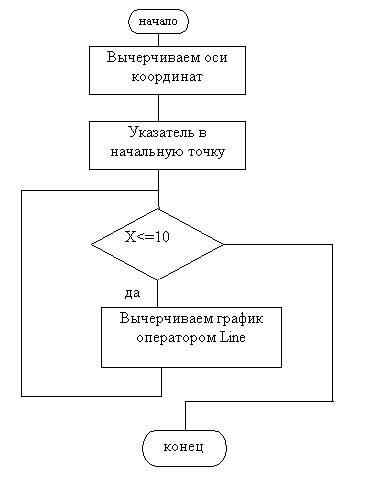


Рис.9. Блок-схема программы

**5.3. Исходный текст программы**

Private Sub Command1\_Click

Picture1.Cls

x0 = 2

y0 = 5.5

Picture1.Line (x0, y0) - (x0 + 8, y0)

Picture1.Print "10"

Picture1.Line (x0, y0 - 4) - (x0, y0 + 6)

Picture1.Print "10"

x = 3

y = -(x + 3) / (x - 2) - y0

Picture1.PSet (x, -y - 6)

Picture1.Print "3"

Picture1.PSet (x, -y)

Do While x <= 10

y = -(x + 3) / (x - 2) - y0

Picture1.Line -(x, -y), RGB(30, 100, 200)

x = (x + 0.1)

Loop

End Sub

Private Sub Command2\_Click

End

End Sub

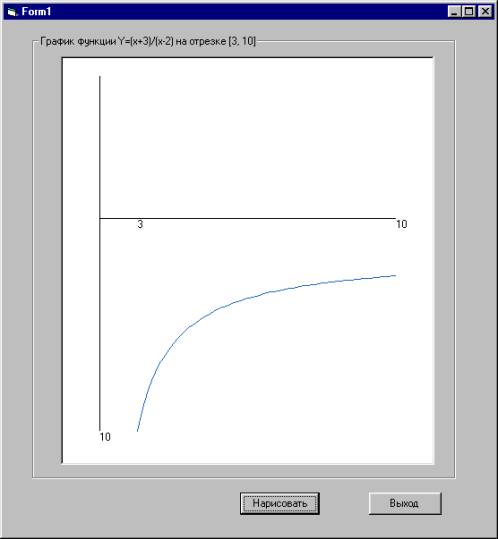


Рис.10. Результат выполнения программы