

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЕВА»
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

**МЕТОДЫ И МАТЕМАТИЧЕСКИЕ МОДЕЛИ
ОПТИМИЗАЦИИ ПРОЕКТНЫХ РЕШЕНИЙ**

ЭЛЕКТРОННЫЙ ВАРИАНТ ЗАДАНИЙ
ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

САМАРА
2010

УДК 629.7.017.1 (075)

Составители: Салмин Вадим Викторович, Старинова Ольга Леонардовна, Петрухина Ксения Вячеславовна.

Даны задания для самостоятельной работы студентов специальности 160400.68 «Ракетные комплексы и космонавтика» направления подготовки по магистерской программе «Проектирование и конструирование космических мониторинговых и транспортных систем», приведен пример выполнения работы.

Разработано на кафедре летательных аппаратов СГАУ.

Содержание

Список заданий для самостоятельной работы.....	4
Пример выполнения работы.....	5

Список заданий для самостоятельной работы

1. Создать программу для решения систем линейных алгебраических уравнений методом Гаусса.
2. Создать программу решения уравнений с одним неизвестным, используя методы Ньютона и парабол.
3. Создать программу, реализующую вычислительный алгоритм метода наискорейшего спуска.
4. Создать программу, реализующую вычислительный алгоритм метода интерполирования функции сплайнами.
5. Создать программу, реализующую вычислительный алгоритм метода сопряженных градиентов.
6. Создать программу, реализующую алгоритм решения краевой задачи методом конечных разностей.
7. Создать программу, реализующую метод динамического программирования в задаче дискретной коррекции движения КА.
8. Создать программу, реализующую синтез оптимального управления линейной динамической системы с квадратичным функционалом.

Пример выполнения работы

Федеральное агентство по образованию
Государственное образовательное учреждение
высшего профессионального образования
**САМАРСКИЙ ГОСУДАРСТВЕННЫЙ
АЭРОКОСМИЧЕСКИЙ УНИВЕРСИТЕТ**
имени академика С.П. КОРОЛЕВА
Кафедра летательных аппаратов

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к профессиональной практике
по численным методам
**ВЫЧИСЛИТЕЛЬНЫЙ АЛГОРИТМ МЕТОДА
«ИНТЕРПОЛИРОВАНИЕ СПЛАЙНАМИ»**

Выполнил студент

группы 1208

Абубекеров А.С.

Проверил _____

Оценка _____

ЗАДАНИЕ

1. Разработать вычислительный алгоритм метода «Интерполирование сплайнами».
2. Разработать программный модуль в среде DELPHI, реализующий данный алгоритм с графической иллюстрацией.
3. Разработать интерфейс пользователя (внутри модуля) и подробное описание работы с ним.
4. Провести тестовый расчет
5. Записать программный продукт на CD
6. Оформить пояснительную записку проекта в соответствии с СТП СГАУ

РЕФЕРАТ

Пояснительная записка 30 страниц, 3 рисунков, 2 источника.

ИНТЕРПОЛИРОВАНИЕ, СПЛАЙН, ПРОИЗВОДНАЯ, МЕТОД ПРОГОНКИ

Цель курсовой работы заключается в разработке программы, реализующей метод интерполирования сплайнами.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	51
ОПИСАНИЕ МЕТОДА.....	6
1.1 Математическая постановка задачи.....	6
1.2 Блок-схема.....	12
1.3 Описание блок-схемы.....	13
2 ОПИСАНИЕ ПРОГРАММЫ.....	14
2.1 Описание переменных и констант.....	14
2.2 Описание модулей.....	15
2.3 Изменение модулей.....	16
3 ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ.....	17
4. ТЕСТОВЫЙ РАСЧЕТ.....	19
4.1 Расчет в системе MathCad.....	19
4.2 Расчет в программе.....	20
ЗАКЛЮЧЕНИЕ.....	22
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	23
ПРИЛОЖЕНИЕ А.....	24

ВВЕДЕНИЕ

Под сплайном (от англ. *spline* — планка, рейка) обычно понимают кусочно-заданную функцию, совпадающую с функциями более простой природы на каждом элементе разбиения своей области определения.

Сплаины имеют многочисленные применения как в математической теории, так и в разнообразных вычислительных приложениях. В частности, сплайны двух переменных интенсивно используются для задания поверхностей в различных системах компьютерного моделирования.

Функции, подобные тем, что сейчас называют сплайнами были известны математикам давно, начиная как минимум с Эйлера, но их интенсивное изучение началось, фактически, только в середине XX века. В 1946 году Исаак Шёнберг впервые употребил этот термин в качестве обозначения класса полиномиальных сплайнов. До 1960 годов сплайны были в основном инструментом теоретических исследований, они часто появлялись в качестве решений различных экстремальных и вариационных задач, особенно в теории приближений.

После 1960 года с развитием вычислительной техники началось использование сплайнов в компьютерной графике и моделировании, что продолжается по сей день.

Целью данной работы является разработка вычислительного алгоритма метода «Интерполирование сплайнами» и разработка программного модуля в среде DELPHI, реализующего данный алгоритм с графической иллюстрацией.

1 ОПИСАНИЕ МЕТОДА

1.1 Математическая постановка задачи

Интерполирование многочленом Лагранжа или Ньютона на отрезке $[a, b]$ с использованием большого числа узлов интерполяции часто приводит к плохому приближению, что объясняется сильным накоплением погрешностей в процессе вычислений. Кроме того из-за расходимости процесса интерполяции увеличение числа узлов не обязательно приводит к повышению точности. Для того, чтобы избежать больших погрешностей, весь отрезок $[a, b]$ разбивают на частичные отрезки и на каждом из частичных отрезков приближенно заменяют функцию многочленом невысокой степени (так называемая кусочно-полиномиальная интерполяция).

Одним из способов интерполяции на всем отрезке является интерполирование с помощью сплайн-функций. Сплайн-функцией или сплайном называют кусочно-полиномиальную функцию, определенную на отрезке $[a, b]$ и имеющую на этом отрезке некоторое число непрерывных производных.

Слово „сплайн” (английское spline) означает гибкую линейку, используемую для проведения гладких кривых через заданные точки плоскости.

Преимущество сплайнов перед обычной интерполяцией является, во-первых, их сходимость, и, во-вторых, устойчивость процесса вычислений.

Рассмотрим частный, но распространенный в вычислительной практике случай, когда сплайн определяется с помощью многочленов третьей степени (кубический сплайн).

Пусть на $[a, b]$ задана непрерывная функция. Введем узлы (сетку):

$$a = x_0 < x_1 < \dots < x_{n-1} < x_n = b,$$

и обозначим $f_i = f(x_i), i \in \overline{0, n}$

Интерполяционным кубическим сплайном, соответствующим данной функции $f(x)$ и данным узлам, называется функция $S(x)$, удовлетворяющая следующим условиям:

а) на каждом сегменте $[x_{i-1}, x_i], i = \overline{1, n}$ функция $S(x)$ является многочленом третьей степени;

б) функция $S(x)$, а так же ее первая и вторая производные непрерывны на $[a, b]$;

$$в) S(x_i) = f(x_i), i = \overline{1, n}$$

Последнее условие называется условием интерполирования.

Докажем существование и единственность сплайна, определяемого перечисленными условиями (плюс некоторые граничные условия, которые будут введены в процессе доказательства). Приводимое ниже доказательство содержит также способ построения сплайна.

На каждом из отрезков $[x_{i-1}, x_i], i = \overline{1, n}$ будем искать функцию $S(x) = S_j(x)$ в виде многочлена третьей степени

$$s_i(x) = a_i + b_i(x - x_i) + \frac{c_i}{2}(x - x_i)^2 + \frac{d_i}{6}(x - x_i)^3, \quad (1)$$

$$x_{i-1} \leq x \leq x_i, \quad i = \overline{1, n},$$

где a_i, b_i, c_i, d_i - коэффициенты, подлежащие определению. Выясним

смысл введенных коэффициентов. Имеем

$$s_i'(x) = b_i + c_i(x - x_i) + \frac{d_i}{2}(x - x_i)^2,$$

$$s_i''(x) = c_i + d_i(x - x_i), \quad s_i'''(x) = d_i,$$

поэтому $a_i = s_i(x_i), \quad b_i = s_i'(x_i), \quad c_i = s_i''(x_i), \quad d_i = s_i'''(x_i).$

Из условий интерполирования $s_i(x_i) = f(x_i), \quad i = \overline{1, n}$, получаем, что

$$a_i = f(x_i), \quad i = \overline{1, n}.$$

Доопределим, кроме того, $a_0 = f(x_0).$

Далее, требование непрерывности функции $s(x)$ приводит к условиям

$$s_i(x_i) = s_{i+1}(x_i), \quad i = \overline{1, n-1}.$$

Отсюда, учитывая выражения для функций $s_i(x)$, получаем при $i = \overline{0, n-1}$ уравнения

$$a_i = a_{i+1} + b_{i+1}(x_i - x_{i+1}) + \frac{c_{i+1}}{2}(x_i - x_{i+1})^2 + \frac{d_{i+1}}{6}(x_i - x_{i+1})^3. \text{ Обоз}$$

начая $h_i = x_i - x_{i-1}$, перепишем эти уравнения в виде

$$h_i b_i - \frac{h_i^2}{2} c_i + \frac{h_i^3}{6} d_i = f_i - f_{i-1}, \quad i = \overline{1, n}. \quad (2)$$

Условия непрерывности первой производной

$$s'_i(x_i) = s'_{i+1}(x_i), \quad i = \overline{1, n-1},$$

приводят к уравнениям

$$c_i h_i - \frac{d_i}{2} h_i^2 = b_i - b_{i-1}, \quad i = \overline{2, n}. \quad (3)$$

Из условий непрерывности второй производной получаем уравнения

$$d_i h_i = c_i - c_{i-1}, \quad i = \overline{2, n}. \quad (4)$$

Объединяя (2)-(4), получим систему $3n - 2$ уравнений относительно $3n$ неизвестных $b_i, c_i, d_i, \quad i = \overline{1, n}$.

Два недостающих условия получают, задавая те или иные граничные условия для $s(x)$. Предположим, например, что функция $f(x)$ удовлетворяет условиям $f''(a) = f''(b) = 0$. Тогда естественно требовать, чтобы

$$s''(a) = s''(b) = 0. \text{ Отсюда получаем } s''_1(x_0) = 0, \quad s''_n(x_n) = 0,$$

$$\text{т.е. } c_1 - d_1 h_1 = 0, \quad c_n = 0.$$

Заметим, что условие $c_1 - d_1 h_1 = 0$ совпадает с уравнением (4) при $i = 1, \quad c_0 = 0$. Таким образом, приходим к замкнутой системе уравнений для определения коэффициентов кубического сплайна:

$$h_i d_i = c_i - c_{i-1}, \quad i = \overline{1, n}, \quad c_0 = c_n = 0, \quad (5)$$

$$h_i c_i - \frac{h_i^2}{2} d_i = b_i - b_{i-1}, \quad i = \overline{2, n}, \quad (6)$$

$$h_i b_i - \frac{h_i^2}{2} c_i + \frac{h_i^3}{6} d_i = f_i - f_{i-1}, \quad i = \overline{1, n}. \quad (7)$$

Убедимся в том, что эта система имеет единственное решение. Исключим из (5)- (7) переменные b_i, d_i , $i = \overline{1, n-1}$, и получим систему, содержащую только c_i , $i = \overline{1, n-1}$. Для этого рассмотрим два соседних уравнения (7) :

$$b_i = \frac{h_i}{2} c_i - \frac{h_i^2}{6} d_i + \frac{f_i - f_{i-1}}{h_i},$$

$$b_{i-1} = \frac{h_{i-1}}{2} c_{i-1} - \frac{h_{i-1}^2}{6} d_{i-1} + \frac{f_{i-1} - f_{i-2}}{h_{i-1}},$$

и вычтем второе уравнение из первого. Тогда получим

$$b_i - b_{i-1} = \frac{1}{2}(h_i c_i - h_{i-1} c_{i-1}) - \frac{1}{6}(h_i^2 d_i - h_{i-1}^2 d_{i-1}) + \frac{f_i - f_{i-1}}{h_i} - \frac{f_{i-1} - f_{i-2}}{h_{i-1}}.$$

Подставляя найденное выражение для $b_i - b_{i-1}$ в правую часть уравнения (6), получим

$$h_i c_i - h_{i-1} c_{i-1} - \frac{h_{i-1}^2}{3} d_{i-1} - \frac{2h_i^2}{3} d_i = 2 \left(\frac{f_i - f_{i-1}}{h_i} - \frac{f_{i-1} - f_{i-2}}{h_{i-1}} \right). \quad (8)$$

Далее, из уравнения (5) получаем

$$h_i^2 d_i = h_i (c_i - c_{i-1}), \quad h_{i-1}^2 d_{i-1} = h_{i-1} (c_{i-1} - c_{i-2}).$$

И подставляя эти выражения в (8), приходим к уравнению

$$h_{i-1} c_{i-2} + 2(h_{i-1} + h_i) c_{i-1} + h_i c_i = 6 \left(\frac{f_i - f_{i-1}}{h_i} - \frac{f_{i-1} - f_{i-2}}{h_{i-1}} \right).$$

Окончательно для определения коэффициентов c_i получаем систему уравнений

$$h_i c_{i-1} + 2(h_i + h_{i+1})c_i + h_{i+1}c_{i+1} = 6 \left(\frac{f_{i+1} - f_i}{h_{i+1}} - \frac{f_i - f_{i-1}}{h_i} \right), \quad (9)$$

$$i = \overline{1, n-1}, \quad c_0 = c_n = 0.$$

В силу диагонального преобладания система (9) имеет единственное решение. Так как матрица системы трехдиагональная, решение можно найти методом прогонки. По найденным коэффициентам c_i коэффициенты b_i и d_i определяются с помощью явных формул

$$d_i = \frac{c_i - c_{i-1}}{h_i}, \quad b_i = \frac{h_i}{2} c_i - \frac{h_i^2}{6} d_i + \frac{f_i - f_{i-1}}{h_i}, \quad (10)$$

$$i = \overline{1, n}.$$

Таким образом, доказано, что существует единственный кубический сплайн, определяемый условиями а)-в) и граничными условиями $s''(a) = s''(b) = 0$. Заметим, что можно рассматривать и другие граничные условия.

1.2 Блок-схема

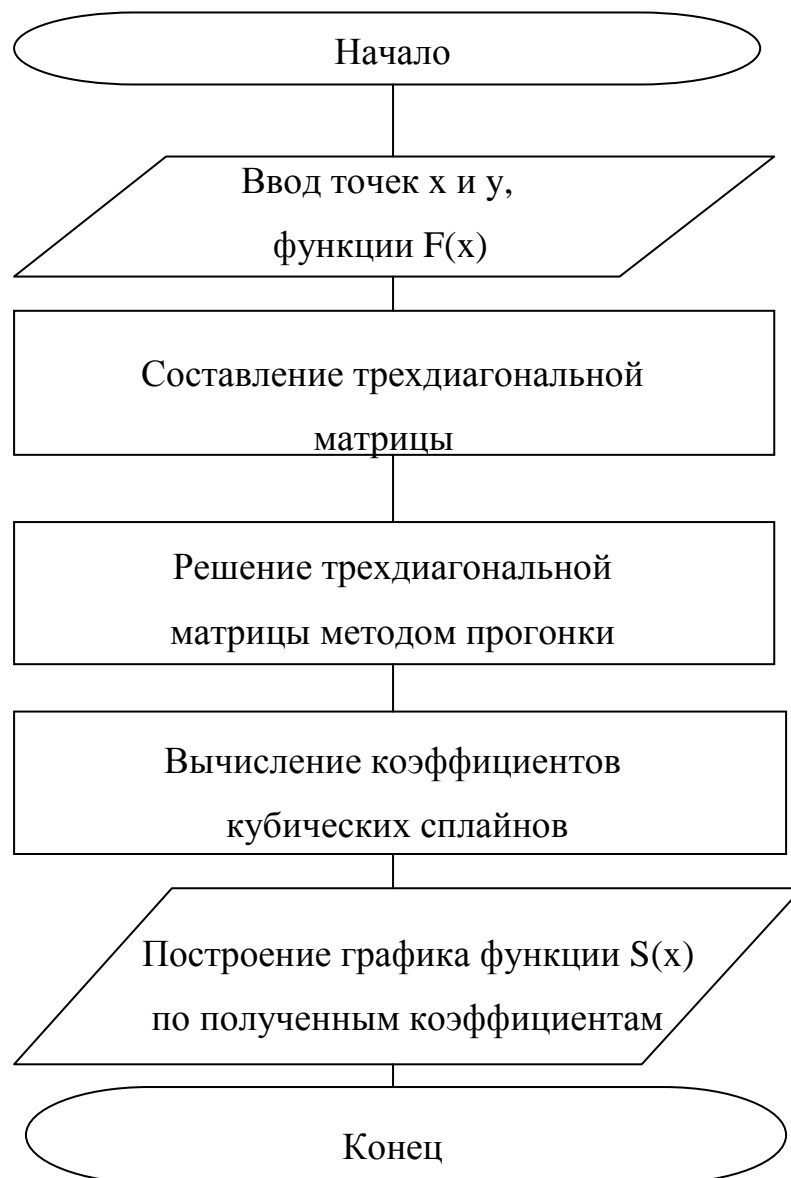


Рисунок 1 – Блок-схема программы

1.3 Описание блок-схемы

Вначале работы программы вводятся координаты точек X, Y интерполируемой функции $f(x)$. Затем составляется функция сплайнов $S(x)$, для определения её коэффициентов необходимо составить трехдиагональную матрицу. Решением этой системы уравнений являются искомые коэффициенты. Далее по явным формулам находятся все остальные необходимые коэффициенты, и составляется функция $S(x)$ которая отображается на графике и является решением задачи.

2. ОПИСАНИЕ ПРОГРАММЫ

2.1 Описание переменных и констант

Основные переменные описаны в таблице 1.

Таблица 1 – описание основных переменных

--	Тип	Описание
x	array [0..1] of real	Массив координат X
y	array [0..1] of real	Массив координат Y
a	array [0..1] of real	Массив коэффициентов a
c	array [0..1] of real	Массив коэффициентов c
alfa	array [0..1-1] of real	Массив коэффициентов α
beta	array [0..1-1] of real	Массив коэффициентов β
hi	real	Разность текущего X и следующего
Hi1	Real	Разность текущего X и следующего
Zk	real	Значение знаменателя для формулы alfa и beta
Ak	real	Коэффициент трехдиагональной матрицы
Bk	real	Коэффициент трехдиагональной матрицы
Ck	real	Коэффициент трехдиагональной матрицы
Fk	real	Коэффициент трехдиагональной матрицы
q	real	Шаг построения графика
S2	real	Значение функции S(x)
b	array [0..1] of real	Массив коэффициентов b
d	array [0..1] of real	Массив коэффициентов d
i	integer	Переменная для циклов
n	integer	Количество элементов массива

Константы описаны в таблице 2.

Таблица 2 – описание констант

1	integer	Количество точек функции F(x)
---	---------	-------------------------------

2.2 Описание модулей

Программа написана на языке программирования Delphi и состоит из 1 модуля Unit 1.

При запуске программы выполняется процедура `FormCreate(Sender: TObject);` в которой устанавливаются начальные значения переменных и свойства объектов.

Затем пользователь должен загрузить значения одной из предложенных функций, за это отвечают процедуры `N3Click(Sender: TObject);`, `N6Click(Sender: TObject);`, `Fxxxx1Click(Sender: TObject);`, `Fxx1cosx1Click(Sender: TObject);`, `Fxxxcosxsinx1Click(Sender: TObject);`, `Fxcosx1Click(Sender: TObject);`; в них выполняются одни и те же действия, различия лишь в том какой файл загрузить.

Вначале очищается поле координат предыдущего графика, очищается поле графика и текстовое поле результатов, затем загружается файл с координатами и заполняется поле координат.

Потом пользователь нажимает кнопку «Интерполирование» за это отвечает процедура `Button1Click(Sender: TObject);` Здесь происходят все вычисления в соответствии с алгоритмом решения задачи. Вначале составляется трехдиагональная матрица, затем она решается методом прогонки. Получив коэффициенты, рассчитываются сплайн функции для каждого участка. В итоге строится график в объекте `TChart` отображающий результат решения задачи.

Так же есть кнопка «Сохранение результатов», которая сохраняет все коэффициенты на каждом участке в файл. За это отвечает процедура `N5Click(Sender: TObject);`

Процедура `N4Click(Sender: TObject);` отвечает за нажатие кнопки справка.

2.3 Изменение модулей

В данной программе есть возможность изменить количество точек, по которым строится график. Достаточно лишь изменить константу L на необходимое значение. Так же пользователь может ввести свои исходные данные, заменив стандартные в текстовых файлах на свои.

3. ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ

После запуска программы на экране появится окно содержащее поле координат, поле графика и результатов. Рисунок

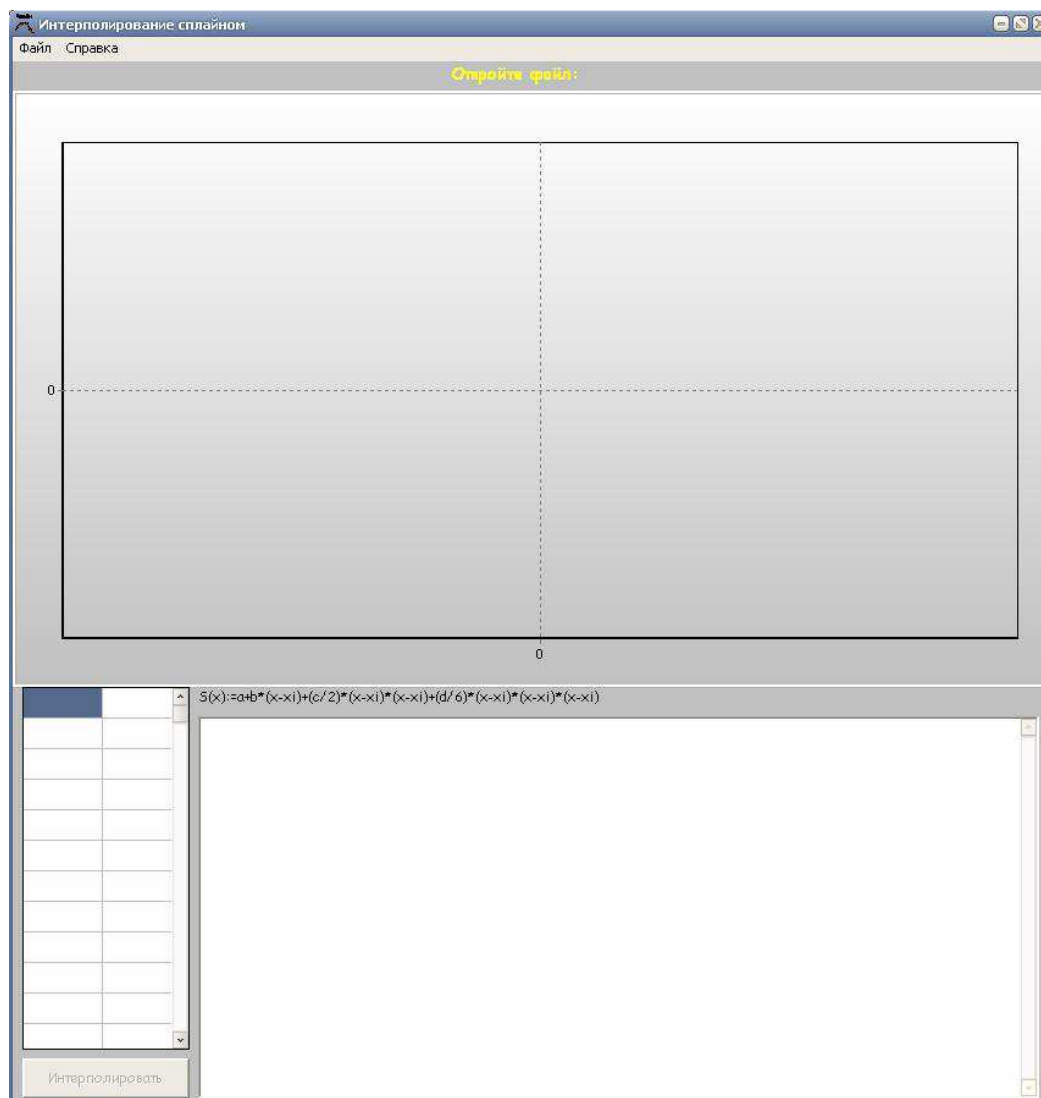


Рисунок 1 – главное окно программы.

Для того чтобы программа начала работать необходимо загрузить координаты точек. Это можно сделать нажав Меню – Файл – Открыть и выбрать одну из функций. После загрузки автоматически заполнится таблица координат точек выбранного графика. После нажатия кнопки «Интерполирование» выполнится расчет всех необходимых коэффициентов для функций сплайнов. Затем отобразится получившийся график интерполирующей сплайн функции и выведутся результаты в текстовое поле.

Все результаты пользователь может сохранить нажав Меню – Файл – Сохранить результаты. Результаты будут сохранены в файл LastResults. Рисунок 2.

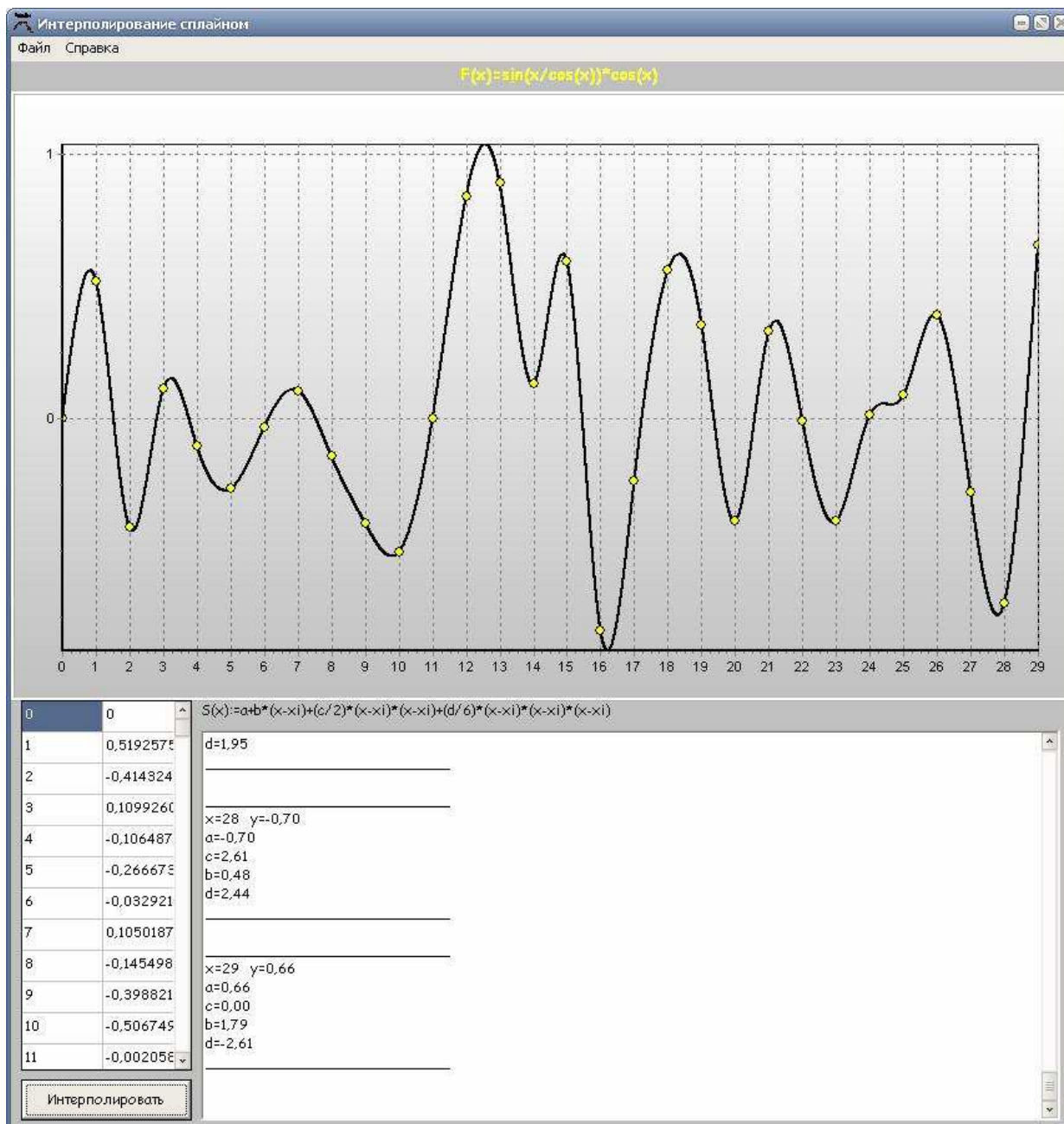


Рисунок 2 – результат работы программы.

4 ТЕСТОВЫЙ РАСЧЕТ

4.1 Расчет в системе MathCad

$$x := \begin{pmatrix} 0 \\ 2 \\ 5 \\ 6 \\ 7 \end{pmatrix} \quad y := \begin{pmatrix} 1 \\ 5 \\ 8 \\ 2 \\ 10 \end{pmatrix}$$

$$i := 0..3$$

$$h_i := x_{i+1} - x_i$$

$$h = \begin{pmatrix} 2 \\ 3 \\ 1 \\ 1 \end{pmatrix}$$

$$j := 0..2 \quad k := 0..1$$

$$M_{j,j} := 2 \cdot (h_j + h_{j+1})$$

$$M_{k+1,k} := h_{k+1}$$

$$M_{k,k+1} := h_{k+1}$$

$$M = \begin{pmatrix} 10 & 3 & 0 \\ 3 & 8 & 1 \\ 0 & 1 & 4 \end{pmatrix} \quad V_j := 6 \cdot \left(\frac{y_{j+2} - y_{j+1}}{h_{j+1}} - \frac{y_{j+1} - y_j}{h_j} \right)$$

$$\alpha_0 := \frac{-M_{0,1}}{M_{0,0}} \quad \beta_0 := \frac{V_0}{M_{0,0}} \quad V = \begin{pmatrix} -6 \\ -42 \\ 84 \end{pmatrix}$$

$$\alpha_1 := \frac{-M_{1,2}}{M_{1,1} + M_{0,1} \cdot \alpha_0} \quad \beta_1 := \frac{-M_{0,1} \cdot \beta_0 + V_1}{M_{1,1} + M_{0,1} \cdot \alpha_0}$$

$$\alpha = \begin{pmatrix} -0.3 \\ -0.141 \end{pmatrix} \quad \beta = \begin{pmatrix} -0.6 \\ -5.662 \end{pmatrix}$$

$$c_0 := 0 \quad c_4 := 0$$

$$c_3 := \frac{-M_{2,1} \cdot \beta_1 + V_2}{M_{2,2} + M_{2,1} \cdot \alpha_1}$$

$$c_2 := \alpha_1 \cdot c_3 + \beta_1$$

$$c_1 := \alpha_0 \cdot c_2 + \beta_0$$

$$d_i := \frac{c_{i+1} - c_i}{h_i} \quad b_i := \frac{h_i \cdot c_{i+1}}{2} - \frac{h_i \cdot h_i \cdot d_i}{6} + \frac{y_{i+1} - y_i}{h_i}$$

$$b = \begin{pmatrix} 3.387 \\ -6.894 \\ 0.255 \\ 11.872 \end{pmatrix} \quad c = \begin{pmatrix} 0 \\ 2.08 \\ -8.934 \\ 23.234 \\ 0 \end{pmatrix} \quad d = \begin{pmatrix} 1.04 \\ -3.672 \\ 32.168 \\ -23.234 \end{pmatrix}$$

4.2 Расчет в программе

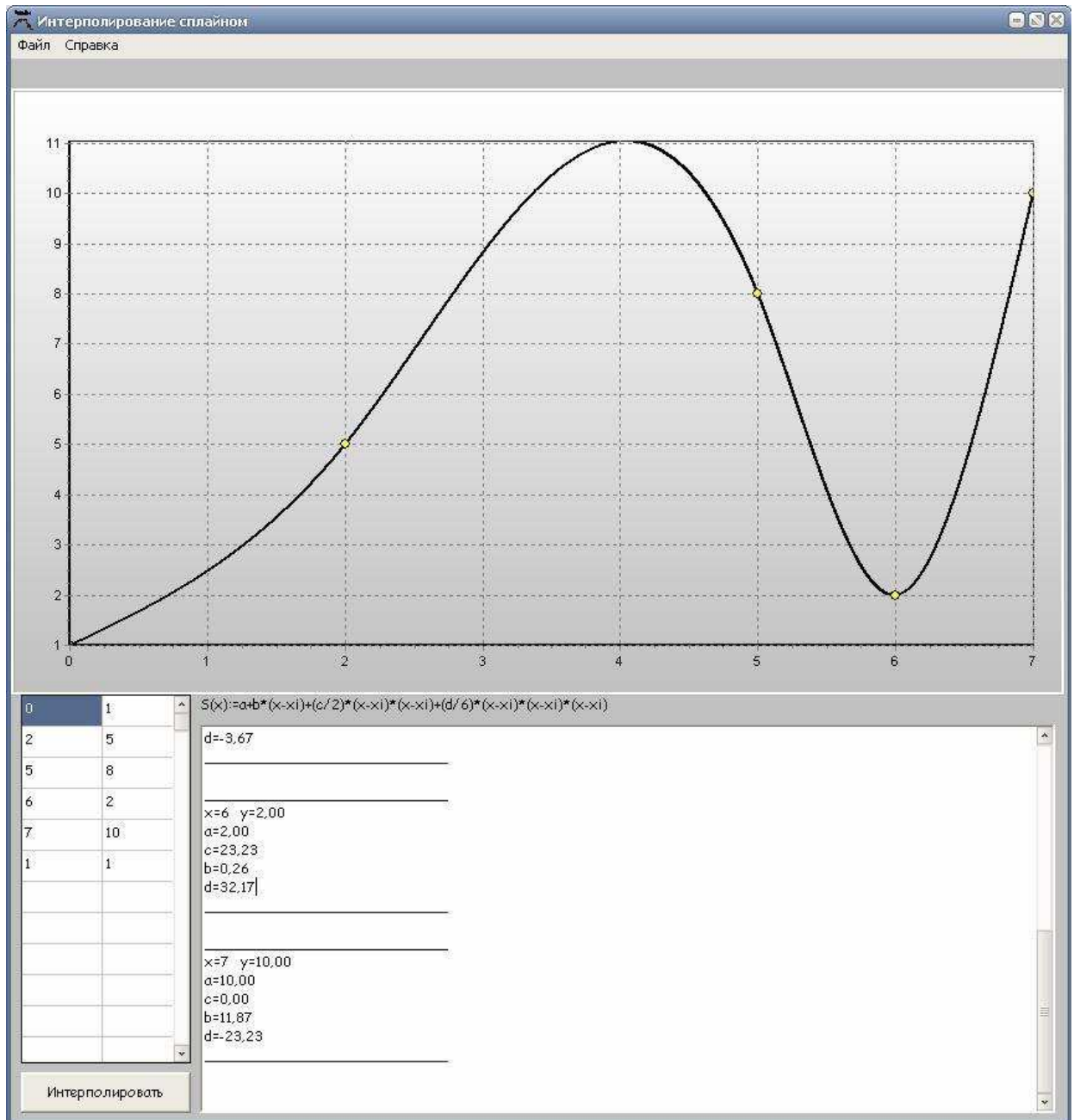


Рисунок 3 – результат работы программы

Содержание файла LastResults.ini:

Коэффициенты результирующей функции S(x):

$$S(x) := a + b(x-x_i) + (c/2)(x-x_i)^2 + (d/6)(x-x_i)^3$$

$$x=0 \quad y=1,00$$

$$a=1,00$$

$$c=0,00$$

$$b=0,00$$

$$d=7,27E303$$

$$x=2 \quad y=5,00$$

$$a=5,00$$

$$c=2,08$$

$$b=3,39$$

$$d=1,04$$

$$x=5 \quad y=8,00$$

$$a=8,00$$

$$c=-8,93$$

$$b=-6,89$$

$$d=-3,67$$

$$x=6 \quad y=2,00$$

$$a=2,00$$

$$c=23,23$$

$$b=0,26$$

$$d=32,17$$

$$x=7 \quad y=10,00$$

$$a=10,00$$

$$c=0,00$$

$$b=11,87$$

$$d=-23,23$$

Результаты, полученные при тестовом расчете в системе MathCad, полностью совпадают с результатом, выполненной программы.

ЗАКЛЮЧЕНИЕ

В данной программе был реализован вычислительный алгоритм метода «Интерполирование сплайнами», разработан программный модуль в среде DELPHI, реализующий данный алгоритм с графической иллюстрацией. Разработан интерфейс пользователя и проведен тестовый расчет. В качестве исходных данных этой программы пользователь мог загружать данные из файлов. Результатом работы программы был график кусочно-полиномиальной функции и таблица коэффициентов каждого сплайна.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Лекционный материал по предмету «Численные методы» СГАУ
3. СТО СГАУ 02068410-004-2007. Общие требования к учебным текстовым документам. Самара 2007

ПРИЛОЖЕНИЕ А

```
unit Unit1;
interface
uses
  math, Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Menus, StdCtrls, TeEngine, Series, ExtCtrls, TeeProcs, Chart,
  Buttons, Grids;
const l=30;
type
  TForm1 = class(TForm)
    grafiki: TChart;
    Button1: TButton;
    Series1: TPointSeries;
    Memo1: TMemo;
    xy: TStringGrid;
    MainMenu1: TMainMenu;
    N1: TMenuItem;
    N2: TMenuItem;
    N3: TMenuItem;
    N4: TMenuItem;
    N5: TMenuItem;
    Label1: TLabel;
    Series2: TLineSeries;
    N6: TMenuItem;
    Fxxxx1: TMenuItem;
    Fxx1cosx1: TMenuItem;
    Fxxxcosxsinx1: TMenuItem;
    Fxcosx1: TMenuItem;
    Label2: TLabel;
  procedure FormCreate(Sender: TObject);
  procedure Button1Click(Sender: TObject);
  procedure N5Click(Sender: TObject);
  procedure N6Click(Sender: TObject);
  procedure Fxxxx1Click(Sender: TObject);
  procedure Fxx1cosx1Click(Sender: TObject);
```

```

    procedure Fxxxcosxsinx1Click(Sender: TObject);
    procedure Fxcosx1Click(Sender: TObject);
    procedure N3Click(Sender: TObject);
private
public
end;
var
    Form1: TForm1;
    x: array [0..1] of real;
    y: array [0..1] of real;
    a: array [0..1] of real;
    c:array [0..1] of real;
    alfa: array [0..1-1] of real;
    betta: array [0..1-1] of real;
    hi:real;
    hi1:real;
    Zk: real;
    Ak: real;
    Bk: real;
    Ck: real;
    Fk: real;
    q : real;
    s2: real;
    b: array [0..1] of real;
    d: array [0..1] of real;
    i,n:integer;

implementation
uses IniFiles;
{$R *.dfm}

procedure TForm1.FormCreate(Sender: TObject);
begin
    n:=1;
end;

```

```

procedure TForm1.Button1Click(Sender: TObject);
begin
  for i:=0 to n do
  begin
    x[i]:=strtofloat(xy.Cells[0,i]);
    y[i]:=strtofloat(xy.Cells[1,i]);
    a[i]:=y[i];
  end;
  c[0]:=0;
  c[n-1]:=0;
  alfa[0]:=0;
  betta[0]:=0;
  for i:=1 to n-1 do
  begin
    hi:=x[i]-x[i-1];
    hi1:=x[i+1]-x[i];
    Ak:=hi;
    Ck:=2*(hi+hi1);
    Bk:=hi1;
    Fk:=6*((y[i + 1] - y[i]) / hi1 - (y[i] - y[i - 1]) / hi);
    Zk:=(Ak * alfa[i - 1] + Ck);
    alfa[i] :=-Bk / zk;
    betta[i] := (Fk - Ak * betta[i - 1]) / zk;
  end;
  // Нахождение решения - обратный ход метода прогонки
  for i:=n-2 downto 0 do c[i]:= alfa[i] * c[i+1] + betta[i];
  // По известным коэффициентам c[i] находим значения b[i] и d[i]
  for i:=n-1 downto 0 do
  begin
    hi:= x[i] - x[i - 1];
    d[i]:=(c[i] - c[i-1]) /hi;
    b[i]:= hi * (2* c[i] + c[i-1]) / 6 + (y[i] - y[i - 1]) / hi;
  end;
  //график

```

```

memo1.Lines.Add('Коэффициенты результирующей функции S(x):');
memo1.Lines.Add('S(x):=a+b*(x-xi)+(c/2)*(x-xi)*(x-xi)+(d/6)*(x-xi)*(x-xi)*(x-xi)');
q:=x[0];
series1.AddXY(x[0],y[0],"");
for i:=0 to n-1 do
begin
memo1.Lines.Add('_____');
memo1.Lines.Add('x='+floattostr(x[i])+' y='+floattostrf(y[i],ffixed,4,2));
memo1.Lines.Add('a='+floattostrf(y[i],ffixed,4,2));
memo1.Lines.Add('c='+floattostrf(c[i],ffixed,4,2));
memo1.Lines.Add('b='+floattostrf(b[i],ffixed,4,2));
memo1.Lines.Add('d='+floattostrf(d[i],ffixed,4,2));
memo1.Lines.Add('_____');
memo1.Lines.Add("");
repeat
s2:=y[i]+b[i]*(q-x[i])+(c[i]/2)*(q-x[i])*(q-x[i])+(d[i]/6)*(q-x[i])*(q-x[i])*(q-x[i]);
series2.AddXY(q,s2,"");
q:=q+0.001;
until (q>=x[i]);
end;
for i:=0 to n-1 do
begin
series1.AddXY(x[i],y[i]);
end;
end;

procedure loadF(j:char);
Var iniFile:TIniFile;
FileName:string;
begin
FileName:='C:\Spline\'+j+'.txt';
try
IniFile:=TIniFile.Create(FileName);
for i:=0 to n do
begin

```

```
    x[i]:=strtofloat(iniFile.ReadString('LastValues','x'+inttostr(i),'1'));
    y[i]:=strtofloat(iniFile.ReadString('LastValues','y'+inttostr(i),'1'));
end;
finally
    iniFile.Free;
end;
end;
```

```
procedure TForm1.N5Click(Sender: TObject);
begin
    Memo1.Lines.SaveToFile('LastResults.ini');
end;
```

```
procedure TForm1.N6Click(Sender: TObject);
begin
    memo1.Clear ;
    series1.Clear ;
    series2.Clear ;
    label2.Caption:=n6.Caption;
    loadf('1');
    for i:=0 to n do
        begin
            xy.cells[0,i]:=floattostr(x[i]);
            xy.cells[1,i]:=floattostr(y[i]);
        end;
    button1.Enabled:=true;
end;
```

```
procedure TForm1.Fxxxx1Click(Sender: TObject);
begin
    memo1.Clear ;
    series1.Clear ;
    series2.Clear ;label2.Caption:=fxxxx1.Caption;
    button1.Enabled:=true;
    loadf('2');
```

```
for i:=0 to n do
begin
  xy.cells[0,i]:=floattostr(x[i]);
  xy.cells[1,i]:=floattostr(y[i]);
end;
end;
```

```
procedure TForm1.Fxx1cosx1Click(Sender: TObject);
begin
  memo1.Clear ;
  series1.Clear ;
  series2.Clear ;label2.Caption:=fxx1cosx1.Caption;
  button1.Enabled:=true;
  loadf('3');
  for i:=0 to n do
  begin
    xy.cells[0,i]:=floattostr(x[i]);
    xy.cells[1,i]:=floattostr(y[i]);
  end;
end;
```

```
procedure TForm1.Fxxxcosxsinx1Click(Sender: TObject);
begin
  memo1.Clear ;
  series1.Clear ;
  series2.Clear ;label2.Caption:=fxxxcosxsinx1.Caption;
  button1.Enabled:=true;
  loadf('4');
  for i:=0 to n do
  begin
    xy.cells[0,i]:=floattostr(x[i]);
    xy.cells[1,i]:=floattostr(y[i]);
  end;
end;
```



```
procedure TForm1.Fxcosx1Click(Sender: TObject);
begin
  memo1.Clear ;
  series1.Clear ;
  series2.Clear ;
  label2.Caption:=fxcosx1.Caption;
  button1.Enabled:=true;
  loadf('5');
  for i:=0 to n do
  begin
    xy.cells[0,i]:=floattostr(x[i]);
    xy.cells[1,i]:=floattostr(y[i]);
  end;
end;
```

```
procedure TForm1.N3Click(Sender: TObject);
begin
  halt;
end;
end.
```