Лабораторная работа №1

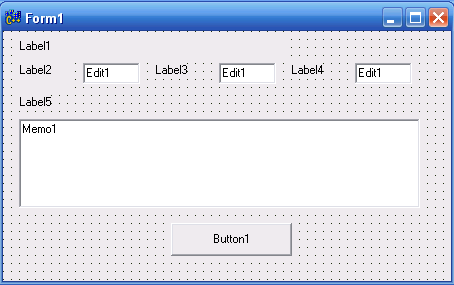
Создать программу вычисления указанной величины. Результат проверить при заданных исходных значениях.



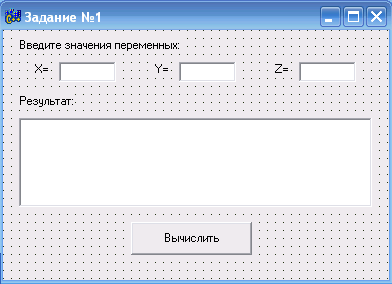
При *x* = –2.235×10-2, *y* = 2.23, *z* = 15.221 :  **39.374**.

**Оконное приложение.**

Создаем окно следующего вида:



Устанавливаем необходимые заголовки:



Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <math.h>

#pragma hdrstop

#include "l1.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

float s;

float x, y, z;

// присваиваем переменным значения из полей

x = StrToFloat(Edit1->Text);

y = StrToFloat(Edit2->Text);

z = StrToFloat(Edit3->Text);

//вычисляем необходимую величину

s = exp(fabs(x-y))\* pow(fabs(x-y),(x+y))/(atan(x)+atan(z))+pow((pow(x,6)+pow(log(y),2)),1/3);

//выводим на экран найденное значение

Memo1->Lines->Add(FloatToStr(s));

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormCreate(TObject \*Sender)

{

Edit1->Text = "-0,02235";

Edit2->Text = "2,23";

Edit3->Text = "15,221";

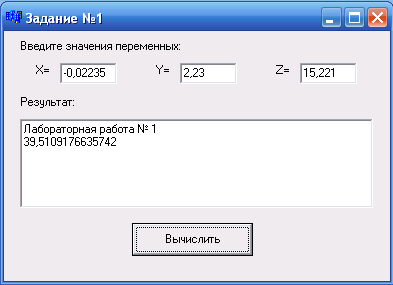
Memo1->Clear();

Memo1->Lines->Add("Лабораторная работа № 1");

}

//---------------------------------------------------------------------------

Результат выполнения программы:



**Консольный режим**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

int main(int argc, char\* argv[])

{

double s;

float x,y,z;

puts("\n\tx,y,z = ");

scanf("%f%f%f", &x, &y, &z);

s = exp(fabs(x-y))\*pow(fabs(x-y),(x+y))/(atan(x)+atan(z))+pow((pow(x,6)+pow(log(y),2)),1/3);

printf("\n x = %7.3f\n y = %7.3f\n z = %7.3f\nRezult = %lf\n", x, y, z, s);

puts("Press any key ... ");

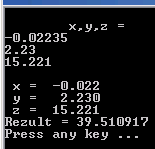
getch();

return 0;

}

//---------------------------------------------------------------------------

Результат выполнения программы:



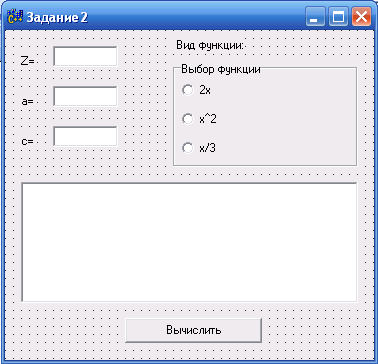
Лабораторная работа №2

Вычислить значение *y* в зависимости от выбранной функции ϕ(*x*), аргумент которой определяется из поставленного условия. Возможные значения функции ϕ(*x*): 2*x*, *x*2, *х*/3 (в оконном приложениии выбор выполнить с помощью компоненты *RadioGroup*). Предусмотреть вывод сообщений, показывающих, при каком условии и с какой функцией производились вычисления *у*.

 где 

**Оконный режим:**

Создаем окно следующего вида:



Код программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <math.h>

#pragma hdrstop

#include "l2.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

float x, y, z, a, c, u;

z = StrToFloat(Edit1->Text);

a = StrToFloat(Edit2->Text);

c = StrToFloat(Edit3->Text);

if(z<0) x = log(z\*z-z);

else x = 2\*z+1;

// Проверка номера нажатой кнопки и выбор соответствующей функции

switch(RadioGroup1->ItemIndex){

case 0: u=2\*x;

Memo1->Lines->Add("2x = "+FloatToStrF(u,ffFixed,8,6));

break;

case 1: u=x\*x;

Memo1->Lines->Add("x^2= "+FloatToStrF(u,ffFixed,8,6));

break;

case 2: u=x/3;

Memo1->Lines->Add("x/3= "+FloatToStrF(u,ffFixed,8,6));

break;

}

//находим значение функции

y=sin(u)\*sin(u)+a\*pow(pow(x,3),5)+c\*log(pow(x,2/5));

//выводим результат

Memo1->Lines->Add("y = "+FloatToStrF(y,ffFixed,8,6));

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormCreate(TObject \*Sender)

{

Edit1->Text="0,1";

Edit2->Text="1";

Edit3->Text="1";

Memo1->Clear();

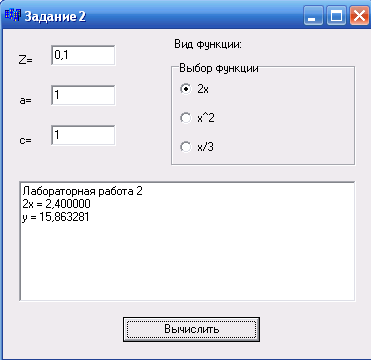
Memo1->Lines->Add("Лабораторная работа 2");

RadioGroup1->ItemIndex = 0;

}

//---------------------------------------------------------------------------

Результат выполнения программы:



**Консольный режим**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <conio.h>

#include <stdio.h>

#include <math.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

int main(int argc, char\* argv[])

{

float x,y,z,a,c,u;

int kod;

puts("Input z, a, c");

scanf("%f%f%f",&z, &a, &c);

if(z<0){ x = log(z\*z-z); printf("z<0");}

else {x = 2\*z+1; printf("z>=0");}

puts("\nFunction U(x):\n 0 - 2x\n 1 - x^2\n 2 - x/3");

scanf("%d", &kod);

switch(kod){

case 0: u=2\*x; printf("\n\t2x= %8.6f",u); break;

case 1: u=x\*x; printf("\n\x^2= %8.6f",u); break;

case 2: u=x/3; printf("\n\x/3= %8.6f",u); break;

}

//находим значение функции

y=sin(u)\*sin(u)+a\*pow(pow(x,3),5)+c\*log(pow(x,2/5));

printf("\nY= %8.6f",y);

puts("\nPress any key ... ");

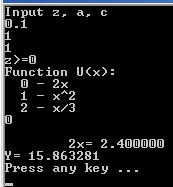
getch();

return 0;

}

//---------------------------------------------------------------------------

Результат выполнения программы:



Лабораторная работа №3

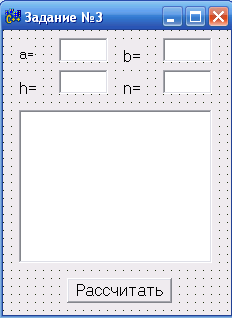
Для каждого ***x***, изменяющегося от ***a*** до ***b*** с шагом ***h***, найти значения функции *Y*(*x*), суммы *S*(*x*) и |*Y*(*x*)–*S*(*x*)| и вывести в виде таблицы. Значения *a*, *b*, *h* и *n* вводятся с клавиатуры. Так как значение *S*(*x*) является рядом разложения функции *Y*(*x*), при правильном решении значения *S* и *Y* для заданного аргумента *x* (для тестовых значений исходных данных) должны совпадать в целой части и в первых двух-четырех позициях после десятичной точки.

Работу программы проверить для *a* = 0,1; *b* = 1,0; *h* = 0,1; значение параметра *n* выбрать в зависимости от задания.

, .

**Оконный режим:**

Создаем окно:



Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <math.h>

#pragma hdrstop

#include "l3.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormCreate(TObject \*Sender)

{

Edit1->Text="0,1"; Edit2->Text="1,0";

Edit5->Text="0,1"; Edit3->Text="10";

Memo1->Lines->Add("Лабораторная работа 3");

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

double a, b, x, h, y, s, r;

int n, k, fact;

a = StrToFloat(Edit1->Text);

b = StrToFloat(Edit2->Text);

n = StrToInt(Edit3->Text);

h = StrToFloat(Edit5->Text);

for(x = a; x<=b; x+=h){

Memo1->Lines->Add("при x= "+FloatToStrF(x,ffFixed,8,2));

y=exp(2\*x);

Memo1->Lines->Add(" Y(x)= "+FloatToStrF(y,ffFixed,8,6));

fact=1; s=0;

for(k=0; k<=n; k++){

if(k==0) fact = 1;

else fact\*=k;

s+= pow(2\*x,k)/fact;

}

Memo1->Lines->Add(" S(x)= "+FloatToStrF(s,ffFixed,8,6));

r=fabs(y-s);

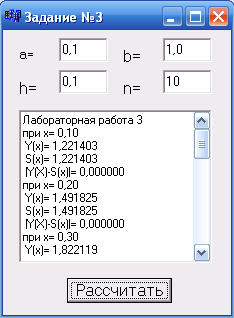
Memo1->Lines->Add(" |Y(X)-S(x)|= "+FloatToStrF(r,ffFixed,8,6));

}

}

//---------------------------------------------------------------------------

Результат выполнения программы:



**Консольное приложение:**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

int main(int argc, char\* argv[])

{

float a, b, x, h, y, s, r;

int n, k, fact;

puts("Input a,b,h,n");

scanf("%f%f%f%d", &a, &b, &h, &n);

for(x = a; x<=b; x+=h){

printf("\n x= %8.2f",x);

y=exp(2\*x);

printf("\n Y(x)= %8.6f",y);

fact=1; s=0;

for(k=0; k<=n; k++){

if(k==0) fact = 1;

else fact\*=k;

s+= pow(2\*x,k)/fact;

}

printf("\n S(x)= %8.6f",s);

r=fabs(y-s);

printf("\n |Y(X)-S(X)|= %8.6f",r);

}

puts("\nPress any key ... ");

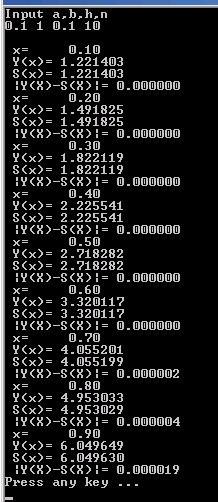
getch();

return 0;

}

//---------------------------------------------------------------------------

Результат выполнения программы:



Лабораторная работа №4

По заданию лабораторной работы №3 написать программу расчета выбранной функции *Y*(*x*), или *S*(*x*) (желательно и |*Y*(*x*)–*S*(*x*)|), вид которой в свою очередь передается в качестве параметра в функцию вывода (*Out*\_*Rez*).

**Оконный режим:**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <math.h>

#pragma hdrstop

#include "l4.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

void Out\_Rez(double, double, double, double, TMemo\*);

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormCreate(TObject \*Sender)

{

Edit1->Text="0,1"; Edit2->Text="1,0";

Edit5->Text="0,1"; Edit3->Text="10";

Memo1->Lines->Add("Лабораторная работа 3");

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

double a, b, x, h, y, s, r;

int n, k, fact;

a = StrToFloat(Edit1->Text);

b = StrToFloat(Edit2->Text);

n = StrToInt(Edit3->Text);

h = StrToFloat(Edit5->Text);

for(x = a; x<=b; x+=h){

y=exp(2\*x);

fact=1; s=0;

for(k=0; k<=n; k++){

if(k==0) fact = 1;

else fact\*=k;

s+= pow(2\*x,k)/fact;

}

r=fabs(y-s);

Out\_Rez(x,y,s,r,Memo1);

}

}

void Out\_Rez(double x,double y,double s,double r, TMemo \*Memo1)

{

Memo1->Lines->Add("при x= "+FloatToStrF(x,ffFixed,8,2));

Memo1->Lines->Add(" Y(x)= "+FloatToStrF(y,ffFixed,8,6));

Memo1->Lines->Add(" S(x)= "+FloatToStrF(s,ffFixed,8,6));

Memo1->Lines->Add(" |Y(X)-S(x)|= "+FloatToStrF(r,ffFixed,8,6));

}

//---------------------------------------------------------------------------

Результат выполнения точно такой как в третьей лабораторной.

**Консольный режим:**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

void Out\_Rez(double, double, double, double);

int main(int argc, char\* argv[])

{

float a, b, x, h, y, s, r;

int n, k, fact;

puts("Input a,b,h,n");

scanf("%f%f%f%d", &a, &b, &h, &n);

for(x = a; x<=b; x+=h){

y=exp(2\*x);

fact=1; s=0;

for(k=0; k<=n; k++){

if(k==0) fact = 1;

else fact\*=k;

s+= pow(2\*x,k)/fact;

}

r=fabs(y-s);

Out\_Rez(x,y,s,r);

}

puts("\nPress any key ... ");

getch();

return 0;

}

void Out\_Rez(double x,double y,double s,double r)

{

printf("\n x= %8.2f",x);

printf("\n Y(x)= %8.6f",y);

printf("\n S(x)= %8.6f",s);

printf("\n |Y(X)-S(X)|= %8.6f",r);

}

//---------------------------------------------------------------------------

Результат выполнения точно такой как в третьей лабораторной.

Лабораторная работа №5

Написать программу по обработке одномерных массивов. Размеры массивов вводить с клавиатуры. В консольном приложении предусмотреть возможность ввода данных как с клавиатуры, так и с использованием функции *random*().

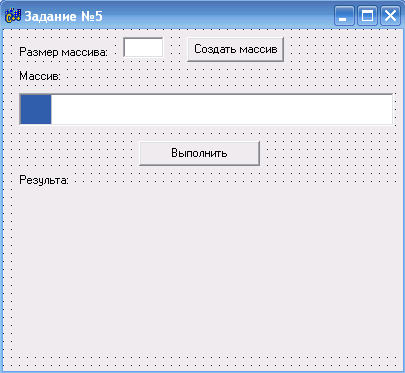
При создании оконного приложения скалярный (простой) результат выводить в виде компоненты *Label*, а массивы вводить и выводить с помощью компонент *StringGrid*.

В одномерном массиве, состоящем из *n* вводимых с клавиатуры целых элементов, вычислить:

Сумму модулей элементов массива, расположенных после последнего отрицательного элемента.

**Оконный режим:**

Создаем окно следующего вида:



Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#pragma hdrstop

#include "l5.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

int n;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormCreate(TObject \*Sender)

{

randomize();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

int i, k=0, s=0, m, a[10];

//заполняем массив

for(i=0; i<n;i++)

a[i]=StrToInt(StringGrid1->Cells[i][0]);

//ищем последний отрицательный

for(i=0;i<n;i++)

if(a[i]<0) k=i;

if(k!=n){

for(i=k+1;i<n;i++)

s+= abs(a[i]);

Label3->Caption="Последний отрицательный находиться на "+IntToStr(k)+" позиции."+

"\nСумму модулей элементов массива, расположенных после последнего\n отрицательного элемента равна "+IntToStr(s);

}

else Label3->Caption="Последний отрицательный находиться на последней позиции";

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

n=StrToInt(Edit1->Text);

StringGrid1->ColCount=n;

StringGrid1->Visible=True;

for(int i=0; i<n;i++) // Заполнение массива А случайными числами

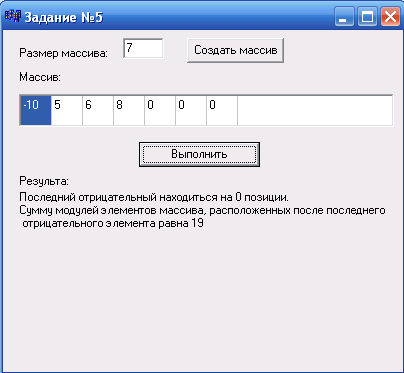
StringGrid1->Cells[i][0] = IntToStr(random(21)-10);

}

//---------------------------------------------------------------------------

Результат выполнения программы:

Когда нажимается кнопка «Создать массив» StringGrid заполняется случайными числами. Для ввода с клавиатуры в StringGrid разрешен ввод.



**Консольное приложение:**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <conio.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

void Rand();

void keyboard();

void output();

int n, a[10], i;

int main(int argc, char\* argv[])

{

int kod, k, s=0;

puts("n= ");

scanf("%d", &n);

puts("\nSposob vvoda:\n 0 - random\n 1 - keyboard\n");

scanf("%d", &kod);

switch(kod){

case 0: Rand(); break;

case 1: keyboard(); break;

}

printf("Massiv: ");

output();

//ищем последний отрицательный

for(i=0;i<n;i++)

if(a[i]<0) k=i;

if(k!=n){

for(i=k+1;i<n;i++)

s+= abs(a[i]);

printf("\n Ickomaja summa= %d",s);

}

else printf("\nPoclednij element v massive otricatelen ");

puts("\nPress any key ... ");

getch();

return 0;

}

void Rand()

{

randomize();

for(i=0;i<n;i++)

a[i]=random(31)-10;

}

void keyboard()

{

puts("\n Vvedite n chicel: ");

for(i=0;i<n;i++)

scanf("%d", &a[i]);

}

void output()

{

for(i=0;i<n;i++)

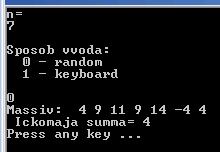
printf(" %d",a[i]);

}

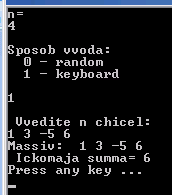
//---------------------------------------------------------------------------

Результат выполнения:

Случайные числа:



Ввод с клавиатуры:



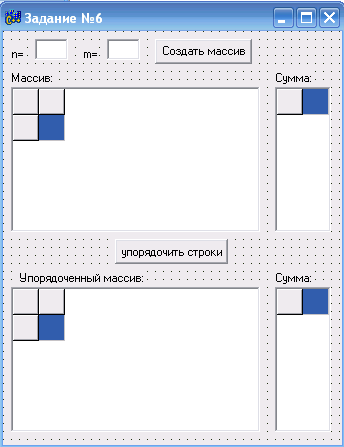
Лабораторная работа №6

Написать программу по обработке динамических массивов. Размеры массивов вводить с клавиатуры. При создании оконного приложения скалярный (простой) результат выводить в виде компоненты *Label*, а массивы вводить и выводить с помощью компонент *StringGrid*, в которых 0-й столбец и 0-ю строку использовать для отображения индексов массивов.

В матрице размером *N*×*M* упорядочить строки по возрастанию суммы их элементов.

**Оконное приложение:**

Создаем окно следующего вида:



Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#pragma hdrstop

#include "l6.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

int n, m;

int \*\*a;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

n=StrToInt(Edit1->Text);

m=StrToInt(Edit2->Text);

StringGrid1->ColCount = m+1;

StringGrid1->RowCount = n+1;

StringGrid1->Visible=true;

StringGrid1->Cells[0][0]="Массив";

for(int i=1; i<=n;i++)

StringGrid1->Cells[0][i]="i="+IntToStr(i);

for(int j=1; j<=m;j++)

StringGrid1->Cells[j][0]="j="+IntToStr(j);

//заполняем случайными числами

randomize();

for(int i=1; i<=n;i++)

for(int j=1; j<=m;j++)

StringGrid1->Cells[j][i]=IntToStr(random(21)-10);

//выделяем память для динамического массива

a = new int\*[n+1];

for(int i=0; i<=n;i++) a[i] = new int[m+1];

//заполняем динамический массив

for(int i=1; i<=n;i++)

for(int j=1; j<=m;j++)

a[i][j]=StrToInt(StringGrid1->Cells[j][i]);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::BitBtn1Click(TObject \*Sender)

{

int \*c, \*y;

//создаем миссив сумм элементов строк и массив индексов

c = new int[n+1];

y = new int[m+1];

for(int j=1; j<=n;j++)

c[j]=0;

for(int i=1; i<=n;i++)

for(int j=1; j<=m;j++)

c[i]+= a[i][j];

StringGrid2->RowCount=n+1;

StringGrid2->Visible=true;

for(int i=1; i<=n;i++){

StringGrid2->Cells[1][i]=IntToStr(c[i]);

StringGrid2->Cells[0][i]="i="+IntToStr(i);

}

//сортируем массив сумм и одновременно матрицу

int x, l, k, i;

for(k=1;k<=n-1;k++){

l=k; x=c[l];

for(i=k+1;i<=n;i++)

if(c[i]<x){

l=i; x=c[l];

for(int j=1; j<=m;j++)

y[j]=a[l][j];

}

c[l]=c[k];

c[k]=x;

for(int j=1; j<=m;j++){

a[l][j]=a[k][j];

a[k][j]=y[j];

}

}

//выводим на экран упорядоченную матрицу

StringGrid3->ColCount = m+1;

StringGrid3->RowCount = n+1;

StringGrid3->Visible=true;

for(int i=1; i<=n;i++)

StringGrid3->Cells[0][i]="i="+IntToStr(i);

for(int j=1; j<=m;j++)

StringGrid3->Cells[j][0]="j="+IntToStr(j);

StringGrid4->RowCount=n+1;

StringGrid4->Visible=true;

for(int i=1; i<=n;i++)

for(int j=1; j<=m;j++)

StringGrid3->Cells[j][i]=IntToStr(a[i][j]);

for(int i=1; i<=n;i++){

StringGrid4->Cells[1][i]=IntToStr(c[i]);

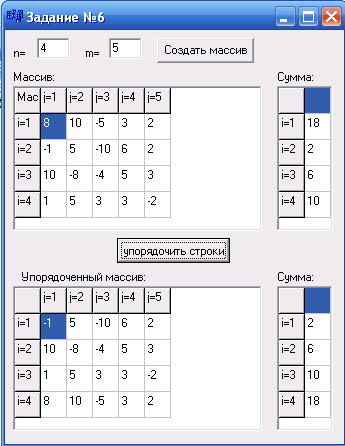
StringGrid4->Cells[0][i]="i="+IntToStr(i);

}

}

//---------------------------------------------------------------------------

Результат работы программы:



**Консольный режим:**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

int n, m;

int \*\*a;

int main(int argc, char\* argv[])

{

printf("Input size N, M : ");

scanf("%d%d",&n,&m);

//выделяем память для динамического массива

a = new int\*[n+1];

for(int i=0; i<=n;i++) a[i] = new int[m+1];

//заполняем динамический массив случайными числами

randomize();

for(int i=1; i<=n;i++)

for(int j=1; j<=m;j++)

a[i][j]=random(21)-10;

//выводим на экран исходную матрицу

printf("Massiv : ");

printf("\n ");

for(int i=1; i<=n;i++){

for(int j=1; j<=m;j++)

printf(" %8.2d",a[i][j]);

printf("\n ");

}

int \*c, \*y;

//создаем миссив сумм элементов строк и массив индексов

c = new int[n+1];

y = new int[m+1];

for(int j=1; j<=n;j++)

c[j]=0;

//подсчитываем суммы

for(int i=1; i<=n;i++)

for(int j=1; j<=m;j++)

c[i]+= a[i][j];

//сортируем массив сумм и одновременно матрицу

int x, l, k, i;

for(k=1;k<=n-1;k++){

l=k; x=c[l];

for(i=k+1;i<=n;i++)

if(c[i]<x){

l=i; x=c[l];

for(int j=1; j<=m;j++)

y[j]=a[l][j];

}

c[l]=c[k];

c[k]=x;

for(int j=1; j<=m;j++){

a[l][j]=a[k][j];

a[k][j]=y[j];

}

}

//выводим на экран упорядоченную матрицу

printf("Sort Massiv : ");

printf("\n ");

for(int i=1; i<=n;i++){

for(int j=1; j<=m;j++)

printf(" %8.2d",a[i][j]);

printf("\n ");

}

puts("\nPress any key ... ");

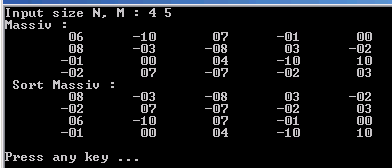
getch();

return 0;

}

//---------------------------------------------------------------------------

Результат работы:



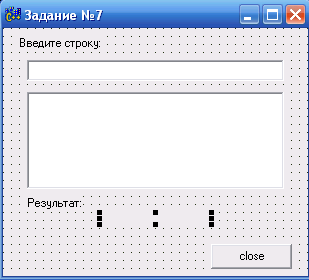
Лабораторная работа №7

В оконном приложении исходные данные вводить из компоненты *Edit* в *ListBox* (заканчивать нажатием *Enter*). Скалярный результат выводить в компоненту *Label*. Для выхода из программы использовать кнопку «*Close*».

Из строки, состоящей из букв, цифр, запятых, точек, знаков + и – , выделить подстроку, задающую вещественное число с плавающей точкой.

**Оконный режим:**

Создаем окно:



Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include<string.h>

#include<stdlib.h>

#pragma hdrstop

#include "l7.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

void cifra(String, int);

String buff = "";

int k;

//создадим массив цифр

String a="0123456789";

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormActivate(TObject \*Sender)

{

Form1->Edit1->SetFocus();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Edit1KeyDown(TObject \*Sender, WORD &Key,

TShiftState Shift)

{

if (Key == 13) {

ListBox1->Items->Add(Edit1->Text);

// Строка из окна редактирования заносится в ComboBox1

Edit1->Text=""; // Очистка окна

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

Form1->Close();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::ListBox1Click(TObject \*Sender)

{

int n, i, j, nst, s=0, l=1;

nst = ListBox1->ItemIndex; // Определение номера строки

String st = ListBox1->Items->Strings[nst]; // Запись выбранной строки st

n=st.Length();

//удаляем все символы до первой цифры

for(i=1;i<=n;i++){

s=0;

for(j=1;j<=10;j++)

if(st[i]==a[j]) s++;

if(s==0){

st.Delete(1,1);

i--;

}

else break;

}

String q=",";

//берем цифры до запятой если она есть

cifra(st,l);

if(st[k]==q[1]){

buff+=st[k]; //присоединяем запятую если она есть

cifra(st,k+1); //продолжаем сбор цифр

}

Label3->Caption=buff;

}

void cifra(String s, int l)

{

int j, c;

for(k=l;k<=s.Length();k++)

{ c=0;

for(j=1;j<=10;j++)

if(s[k]==a[j])c++;

if(c!=0)buff+=s[k];

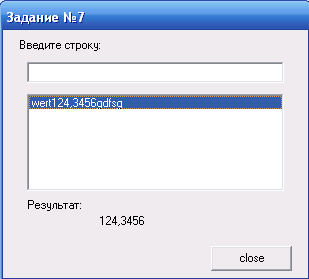
else break;

}

}

//---------------------------------------------------------------------------

Результат работы программы:



**Консольное приложение:**

//---------------------------------------------------------------------------

#include <vcl.h>

#include<string.h>

#include<stdlib.h>

#include <stdio.h>

#include <conio.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

void cifra(String, int);

String buff = "";

int k;

//создадим массив цифр

String a="0123456789";

int main(int argc, char\* argv[])

{

char ss[50];

String st="";

int n, i, s, j, l=1;

printf(" Input string ");

scanf("%s",ss);

st=ss;

n=st.Length();

//удаляем все символы до первой цифры

for(i=1;i<=50;i++){

s=0;

for(j=1;j<=10;j++)

if(st[i]==a[j]) s++;

if(s==0){

st.Delete(1,1);

i--;

}

else break;

}

String q=",";

//берем цифры до запятой если она есть

cifra(st,l);

if(st[k]==q[1]){

buff+=st[k]; //присоединяем запятую если она есть

cifra(st,k+1); //продолжаем сбор цифр

}

printf("\nChiclo %s",buff);

puts("\nPress any key ... ");

getch();

return 0;

}

void cifra(String s, int l)

{

int j, c;

for(k=l;k<=s.Length();k++)

{ c=0;

for(j=1;j<=10;j++)

if(s[k]==a[j])c++;

if(c!=0)buff+=s[k];

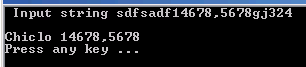
else break;

}

}

//---------------------------------------------------------------------------

Результат работы программы:



Лабораторная работа №8

Написать программу обработки файла типа запись, содержащую следующие пункты меню: «Создание», «Просмотр», «Коррекция» (добавление новых данных или редактирование старых), «Решение индивидуального задания».

Каждая запись должна содержать следующую информацию о студентах:

– фамилия и инициалы;

– год рождения;

– номер группы;

– оценки за семестр: по физике, математике, информатике, химии;

– средний балл.

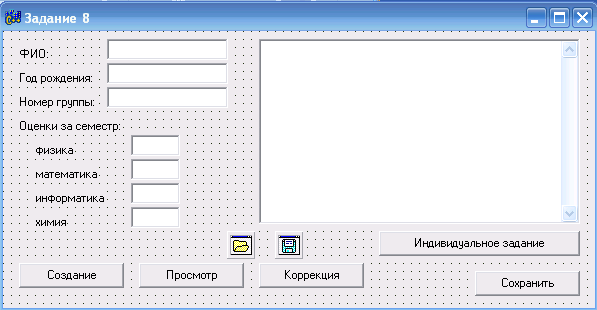
Организовать ввод исходных данных, средний балл рассчитать по введенным оценкам.

Содержимое всего файла и результаты решения индивидувльного задания записать в текстовый файл.

Вычислить общий средний балл всех студентов и распечатать список студентов со средним баллом выше общего среднего балла.

**Оконный режим:**

Создаем окно:



Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <io.h>

#pragma hdrstop

#include "l8.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

struct person{

char FIO[30];

int god;

int num;

int fiz;

int math;

int inf;

int xim;

float ball;

};

int size = sizeof(person);

FILE\* Fz;

AnsiString File\_Zap;

person stud;

person \*group;

void Out(person, TMemo\*);

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::FormCreate(TObject \*Sender)

{

Edit1->Text=""; Edit2->Text="";

Edit3->Text=""; Edit4->Text="";

Edit5->Text=""; Edit6->Text="";

Edit7->Text="";

Memo1->Clear();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

OpenDialog1->Title="Создать новый файл";

if (OpenDialog1->Execute()){

File\_Zap = OpenDialog1->FileName;

if ((Fz=fopen(File\_Zap.c\_str(),"wb"))==NULL) {

ShowMessage("Ошибка создания ФАЙЛА!");

return;

}

}

Memo1->Lines->Add("Создали новый файл: "+AnsiString(File\_Zap));

fclose(Fz);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

Fz = fopen(File\_Zap.c\_str(),"ab");

strcpy(stud.FIO, Edit1 -> Text.c\_str());

stud.god=StrToInt(Edit2->Text);

stud.num=StrToInt(Edit3->Text);

stud.fiz=StrToInt(Edit4->Text);

stud.math=StrToInt(Edit5->Text);

stud.inf=StrToInt(Edit6->Text);

stud.xim=StrToInt(Edit7->Text);

stud.ball=(stud.fiz+stud.math+stud.inf+stud.xim)/4;

Out(stud, Memo1);

fwrite(&stud, size, 1, Fz);

Edit1->Text=""; Edit2->Text="";

Edit3->Text=""; Edit4->Text="";

Edit5->Text=""; Edit6->Text="";

Edit7->Text="";

fclose(Fz);

}

void Out(person z, TMemo \*Memo1)

{

Memo1->Lines->Add("Фамилия "+AnsiString(z.FIO));

Memo1->Lines->Add("Год рождения "+AnsiString(z.god));

Memo1->Lines->Add("Номер группы "+AnsiString(z.num));

Memo1->Lines->Add("Физика "+AnsiString(z.fiz));

Memo1->Lines->Add("Математика "+AnsiString(z.math));

Memo1->Lines->Add("Информатика "+AnsiString(z.inf));

Memo1->Lines->Add("Химия "+AnsiString(z.xim));

Memo1->Lines->Add("Средний балл "+AnsiString(z.ball));

Memo1->Lines->Add("------------");

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

OpenDialog1->Title="Открыть файл";

if (OpenDialog1->Execute()) {

File\_Zap = OpenDialog1->FileName;

if ((Fz=fopen(File\_Zap.c\_str(),"rb"))==NULL) {

ShowMessage("Ошибка открытия ФАЙЛА!");

return;

}

}

Memo1->Lines->Add("--------- Все сведения ---------");

while(1){

if(!fread(&stud,size,1,Fz)) break;

Out(stud, Memo1);

}

fclose(Fz);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button4Click(TObject \*Sender)

{

Fz = fopen(File\_Zap.c\_str(),"rb");

int D\_f = fileno(Fz); // Находим дескриптор файла

int len = filelength(D\_f); // Находим размер файла

int i, j, kol;

kol = len/size; //Количество записей в файле

group = new person[kol];

// Считываем записи из файла в динамический массив

for (i=0; i < kol; i++)

fread((group+i), size, 1, Fz);

fclose(Fz);

float sred; int sum=0;

//находим общий средний балл

for (i=0; i < kol; i++)

sum+=group[i].ball;

sred=sum/kol;

Memo1->Lines->Add("Общий средний балл: "+FloatToStr(sred));

for (i=0; i < kol; i++)

if(group[i].ball>sred)

{

Memo1->Lines->Add("Фамилия: "+AnsiString(group[i].FIO));

Memo1->Lines->Add("Средний балл: "+FloatToStr(group[i].ball));

Memo1->Lines->Add("---------------");

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button5Click(TObject \*Sender)

{

SaveDialog1->Title="Сохранить файл результатов";

if (SaveDialog1->Execute()) {

AnsiString FileNameRez = SaveDialog1->FileName;

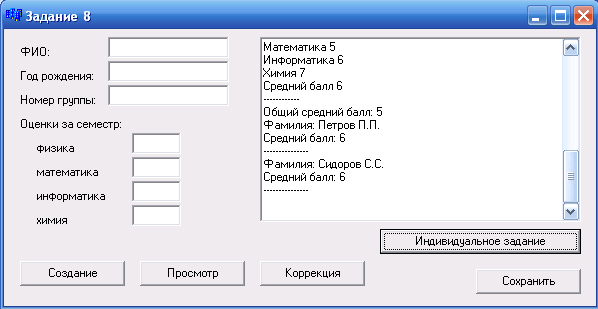
Memo1->Lines->SaveToFile(FileNameRez);

}

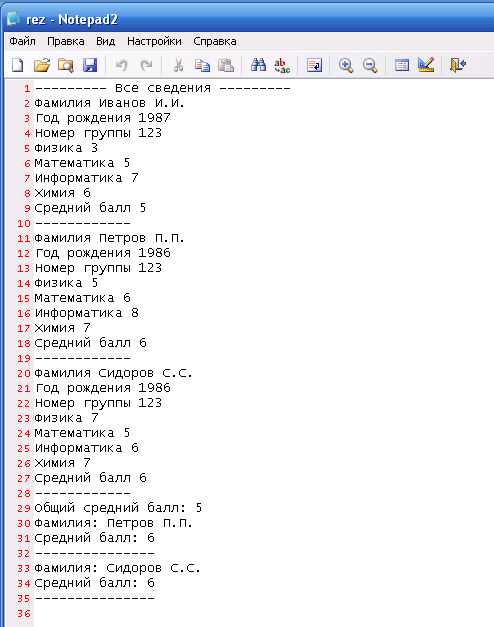
}

//---------------------------------------------------------------------------

Результат выполнения программы:



Результирующий файл:



**Консольный режим:**

Текст программы:

//---------------------------------------------------------------------------

#include <vcl.h>

#include <stdio.h>

#include <io.h>

#pragma hdrstop

//---------------------------------------------------------------------------

#pragma argsused

struct person{

char FIO[30];

int god;

int num;

int fiz;

int math;

int inf;

int xim;

float ball;

};

int size = sizeof(person);

FILE \*Fz, \*Ft;

char File\_Zap[] = "zapisi.dat";

char File\_Rez[] = "rezult.txt";

person stud;

void Out(person);

void main(int argc, char\* argv[])

{

int kod, D\_f, i=0, j, kol;

long len;

person st, \*mas\_Z;

Ft = fopen(File\_Rez, "w");

while(true) {

puts("\n Create - 1\n Add - 2\n View - 3\n Individ\_zad - 4\n EXIT - 0");

scanf("%d", &kod);

switch(kod) {

case 0:

fclose(Ft);

case 1:

if ((Fz=fopen(File\_Zap,"wb"))==NULL) {

puts("\n Create ERROR!");

return;

}

fclose(Fz);

printf("\n Create New File %s !\n",File\_Zap);

break;

case 2:

Fz = fopen(File\_Zap,"ab");

printf("\n F.I.O. - ");

fflush(stdin);

gets(stud.FIO);

printf(" God - ");

scanf("%d", &stud.god);

printf(" Num - ");

scanf("%d", &stud.num);

printf(" Fizika - ");

scanf("%d", &stud.fiz);

printf(" Mathematika - ");

scanf("%d", &stud.math);

printf(" Inform - ");

scanf("%d", &stud.inf);

printf(" Ximija - ");

scanf("%d", &stud.xim);

stud.ball=(stud.fiz+stud.math+stud.inf+stud.xim)/4;

fwrite(&stud, size, 1, Fz);

fclose(Fz);

break;

case 3:

if ((Fz=fopen(File\_Zap,"rb"))==NULL) {

puts("\n Open ERROR!");

return;

}

// Вывод на экран

printf("\n\t--------- Informations ---------");

// Запись такой же информации в текстовый файл Ft

fprintf(Ft,"\n\t--------- Informations ---------");

while(1) {

if(!fread(&stud,size,1,Fz)) break;

Out(stud);

}

fclose(Fz);

break;

case 4:

Fz = fopen(File\_Zap,"rb");

D\_f = fileno(Fz);

len = filelength(D\_f);

kol = len/size;

mas\_Z = new person[kol];

// Считываем записи из файла в динамический массив

for (i=0; i < kol; i++)

fread((mas\_Z+i), size, 1, Fz);

fclose(Fz);

printf("\n\t----- Individ\_zad -----\n");

fprintf(Ft,"\n\t----- Individ\_zad -----\n");

float sred; int sum=0;

//находим общий средний балл

for (i=0; i < kol; i++)

sum+=mas\_Z[i].ball;

sred=sum/kol;

printf("\n\tОбщий средний балл: %f", sred);

fprintf(Ft,"\n\tОбщий средний балл: %f", sred);

for (i=0; i < kol; i++)

if(mas\_Z[i].ball>sred)

{

printf("\n\tФамилия: %s", mas\_Z[i].FIO);

fprintf(Ft,"\n\tФамилия: %s", mas\_Z[i].FIO);

printf("\n\tСредний балл: %d", mas\_Z[i].ball);

fprintf(Ft,"\n\tСредний балл: %d", mas\_Z[i].ball);

printf("\n\t---------\n");

fprintf(Ft,"\n\t--------\n");

}

break;

return;

}

}

}

void Out(person z)

{

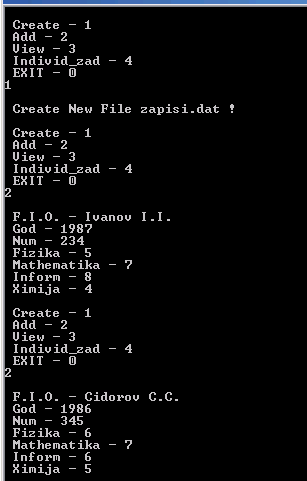
printf("\n %20s , %6.3d, %6.3d, %6.3d, %6.3d, %6.3d, %6.3d, %6.3d .", stud.FIO,stud.god,stud.num,stud.fiz,stud.fiz,stud.math,stud.inf,stud.xim,stud.ball);

fprintf(Ft, "\n %20s , %6.3d, %6.3d, %6.3d, %6.3d, %6.3d, %6.3d, %6.3d .", stud.FIO,stud.god,stud.num,stud.fiz,stud.fiz,stud.math,stud.inf,stud.xim,stud.ball);

}

//---------------------------------------------------------------------------

Результат выполнения:



Результирующий файл:

