**// Указатели.**

#include <iostream>

using namespace std;

int main()

{

int v00=2;

int \*pv00=&v00; // переменная pv00 типа указатель. Содержит адрес перемнной типа int

cout <<" v00= "<<v00<< " address v00 = " << &v00 <<" " <<pv00<< endl;

double v01;

double \*pv01=&v01;

v01=\*pv00+1;

cout <<" v01= "<<v01<<" address v01 = " << &v01 << endl;

cout <<" v00= "<< \*pv00<<" size of v00 = " << sizeof (v00) << endl;

cout <<" v01= "<< \*pv01<<" size of v01 = " << sizeof \*pv01 << endl;

cout << " v01+v02 = " << \*pv00+\*pv01 << endl;

\*pv00= \*pv00+1;

\*pv01= \*pv01+1;

cout <<" new v00= "<<v00 << endl;

cout <<" new v01= "<<v01 << endl;

cout << " new v01+v02 = " << \*pv00+\*pv01 << endl;

int \*pvalue1 = new int; // динамическое выделение памяти под объект типа int

\*pvalue1 = 9; // инициализация объекта через указатель

int \*pvalue2 = new int (10); // инициализация может выполнятся

// сразу при объявлении динамического объекта

cout << "value1 = " << \*pvalue1 << endl; // Разыменование и вывод

cout << "value2 = " << \*pvalue2 << endl;

cout << " address value1 = " << pvalue1 << endl; //вывод адреса

cout << " address value2 = " << pvalue2 << endl;

cout << " value1+value2 = " << \*pvalue1+\*pvalue2 << endl;

delete pvalue1;

delete pvalue2; // высвобождение памяти

double b[5]={0,1,2,3,4};

double\* pb=new double[5];

pb=b; // имя массива определяет адрес

cout<<" 1\_adress b[0] :"<<pb<<" b[0]="<<\*pb<<endl;

pb=&b[0]; // второй способ получения адреса массива

cout<<" 2\_adress b[0] :"<<pb<<" b[0]="<<\*pb<<endl;

pb=pb+1;

cout<<" \_adress b[1] :"<<pb<<" b[1]="<<\*pb<<endl;

pb=pb+1;

cout<<" \_adress b[2] :"<<pb<<" b[2]="<<\*pb<<endl;

pb=pb+2;

cout<<" \_adress b[4] :"<<pb<<" b[4]="<<\*pb<<endl;

pb=pb-1;

cout<<" \_adress b[3] :"<<pb<<" b[3]="<<\*pb<<endl;

delete [] pb;

//Поиск суммы элементов динамического массива вещественных чисел

int i,n; // переменные для индекса и размера массива

float \*pa; // указателя на «float»

float s; //переменная для суммы

cout<<"n=";

cin>>n; //ввод размера массива

//выделение памяти для динамического массива

pa=new float[n];

cout << "vvedite massiv A\n";

for (i=0;i<n; i++) //последовательный ввод элементов массива

//cin>>a[i]; //можно и так

cin>>\*(pa+i);

for(s=0, i=0; i<n; i++) //вычисление суммы

//s+=a[i];

s+=\*(pa+i);

cout << "S= "<<s<<'\n'; //вывод результата

//освобождение памяти

delete [] pa;

return 0;

}