

Обработка одномерных целочисленных массивов на языке программирования Паскаль (программное обеспечение: Pascal ABC).

Определения:

Массив — это поименованная совокупность однотипных элементов, упорядоченных по индексам, определяющим положение элементов в массиве. В языках программирования массивы используются для реализации таких структур данных, как последовательности и таблицы.

Перед использованием в программе массив должен быть описан.

Общий вид описания одномерного массива:

var

```
<имя_массива>: array [<мин_знач_индекса>..of тип_элементов;
```

Пример:

var

```
m: array [1 .. 15] of integer;
```

{Здесь описан массив с именем *m* , состоящий из **15** элементов, каждый элемент – **целое число**}

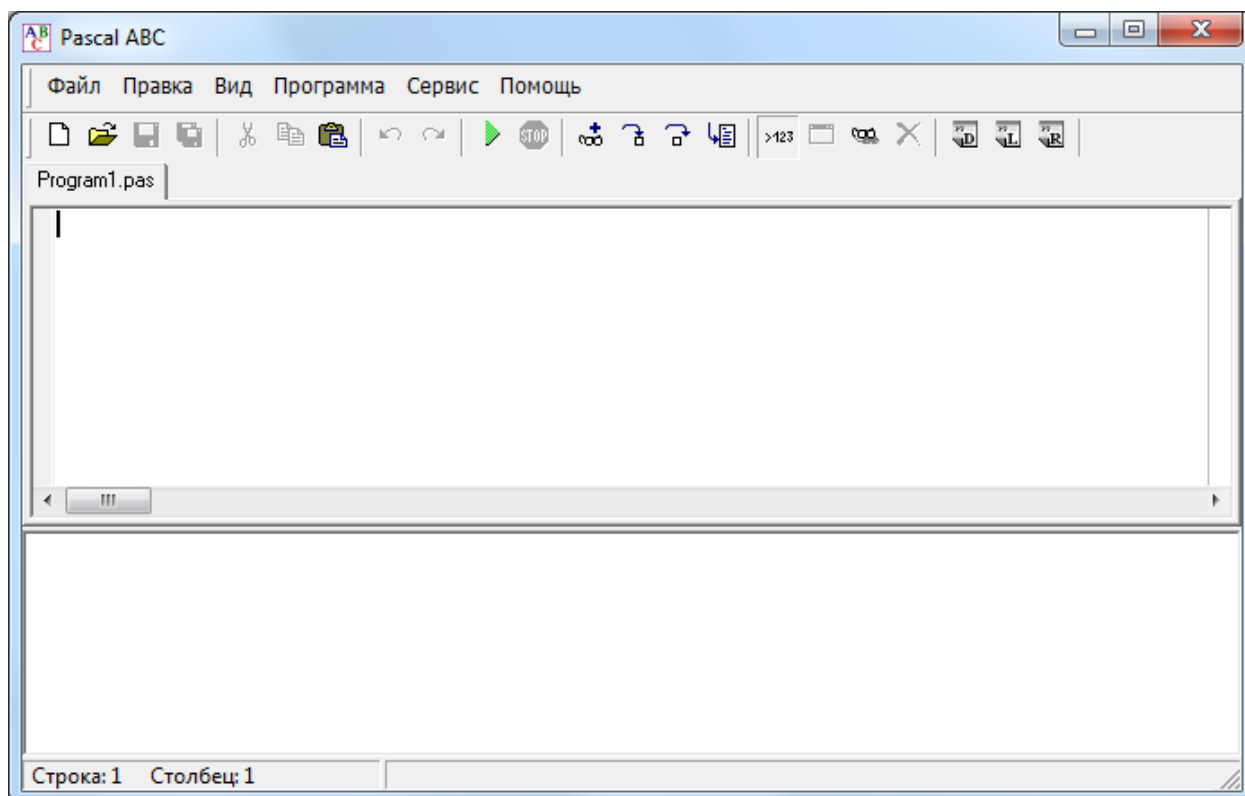
Заполнять массив можно либо вводя значение каждого элемента с клавиатуры, либо присваивая элементам некоторые значения в программе. При заполнении массива и его выводе на экран используется цикл с параметром.

При решении разнообразных задач, связанных с обработкой массивов, используются такие типовые алгоритмы, как:

- суммирование элементов массива;
- поиск элемента с заданными свойствами;
- сортировка массива.

Рассмотрим реализацию этих задач с использованием языка программирования Паскаль и программного обеспечения – среды Pascal ABC.

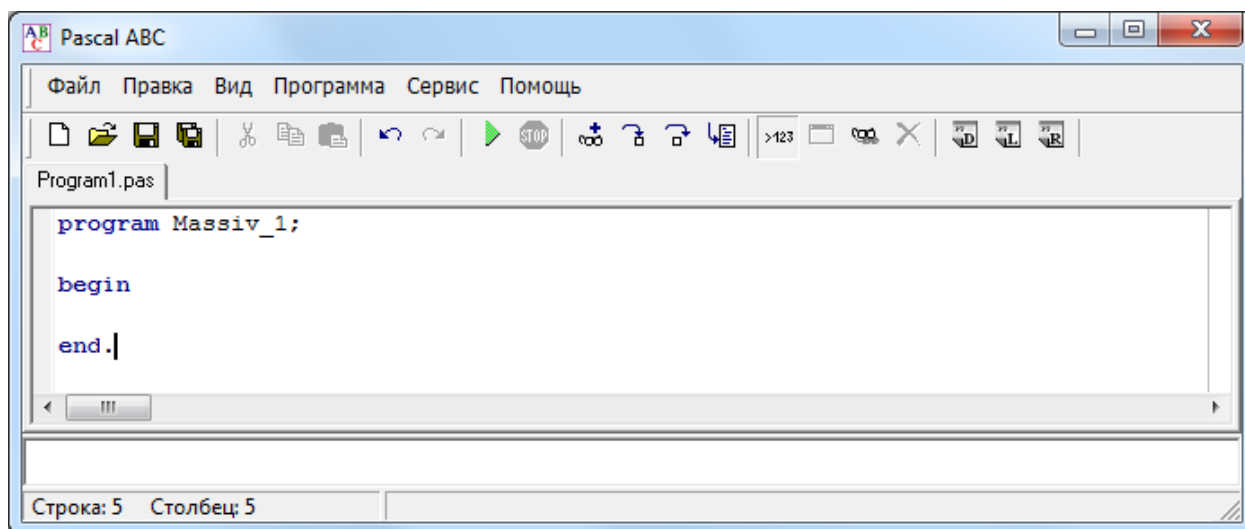
Запустим программу командой: **Все программы – Pascal ABC - Pascal ABC**



Откроется окно программы и автоматически будет создан файл с именем по умолчанию – *Program1.pas* (при сохранении программ рекомендуется давать собственные имена файлов).

После этого можно вводить текст программы, которую после редактирования и отладки нужно запустить на исполнение, ввести исходные (тестовые) данные и проанализировать результат.

Если результаты совпадают с тестовыми данными – программа работает правильно.



{Пример редактирования текста программы – введено тело программы}

Пример 1. Ввод и вывод одномерного целочисленного массива.

Запишем программу, в которой осуществляется: заполнение целочисленного массива **a** ручным вводом с клавиатуры, состоящего из 7 элементов, целыми числами, значения которых заданы в таблице и вывод массива **a** на экран.

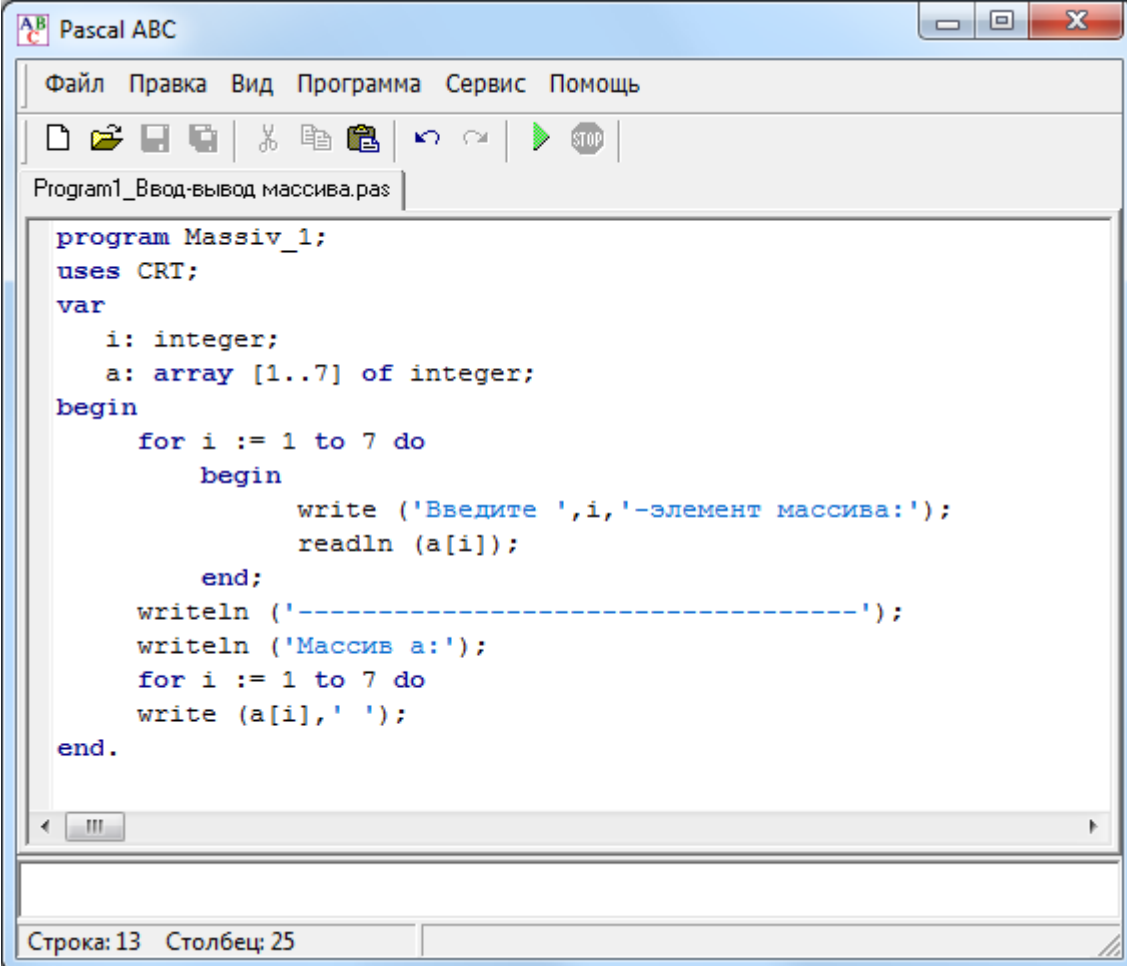
При этом, для удобства работы с программой, будем использовать модуль **CRT** (модуль *CRT* предназначен для работы с текстовым окном; подключается командой: `uses CRT`).

Входные данные:

массив **a**

5	8	12	31	4	-3	1
---	---	----	----	---	----	---

Текст программы:

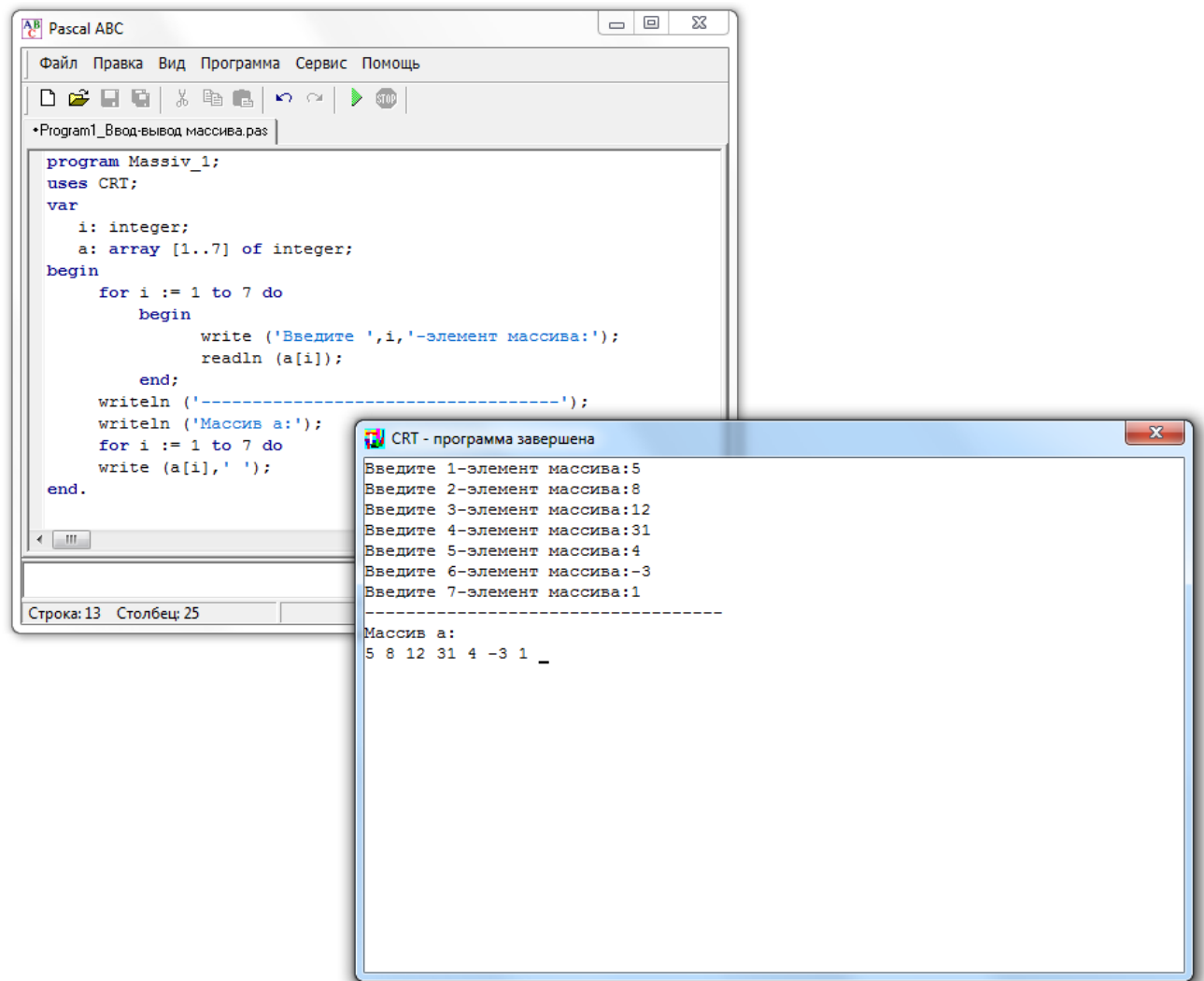


The screenshot shows the Pascal ABC IDE window titled "Pascal ABC". The menu bar includes "Файл", "Правка", "Вид", "Программа", "Сервис", and "Помощь". The toolbar contains icons for file operations, editing, and execution. The active file is "Program1_Ввод-вывод массива.pas". The code in the editor is as follows:

```
program Massiv_1;
uses CRT;
var
  i: integer;
  a: array [1..7] of integer;
begin
  for i := 1 to 7 do
  begin
    write ('Введите ', i, '-элемент массива:');
    readln (a[i]);
  end;
  writeln ('-----');
  writeln ('Массив a:');
  for i := 1 to 7 do
    write (a[i], ' ');
  end.
end.
```

The status bar at the bottom indicates "Строка: 13" and "Столбец: 25".

Выполнение программы:



Массив возможно вводить и без дополнительных комментариев, тогда цикл ввода будет иметь более простой вид:

```
for i := 1 to 7 do
  readln (a[i]);
```

Пример 2. Нахождение суммы элементов одномерного целочисленного массива.

Запишем программу, в которой осуществляется нахождение суммы элементов целочисленного массива **a**, состоящего из 5 элементов.

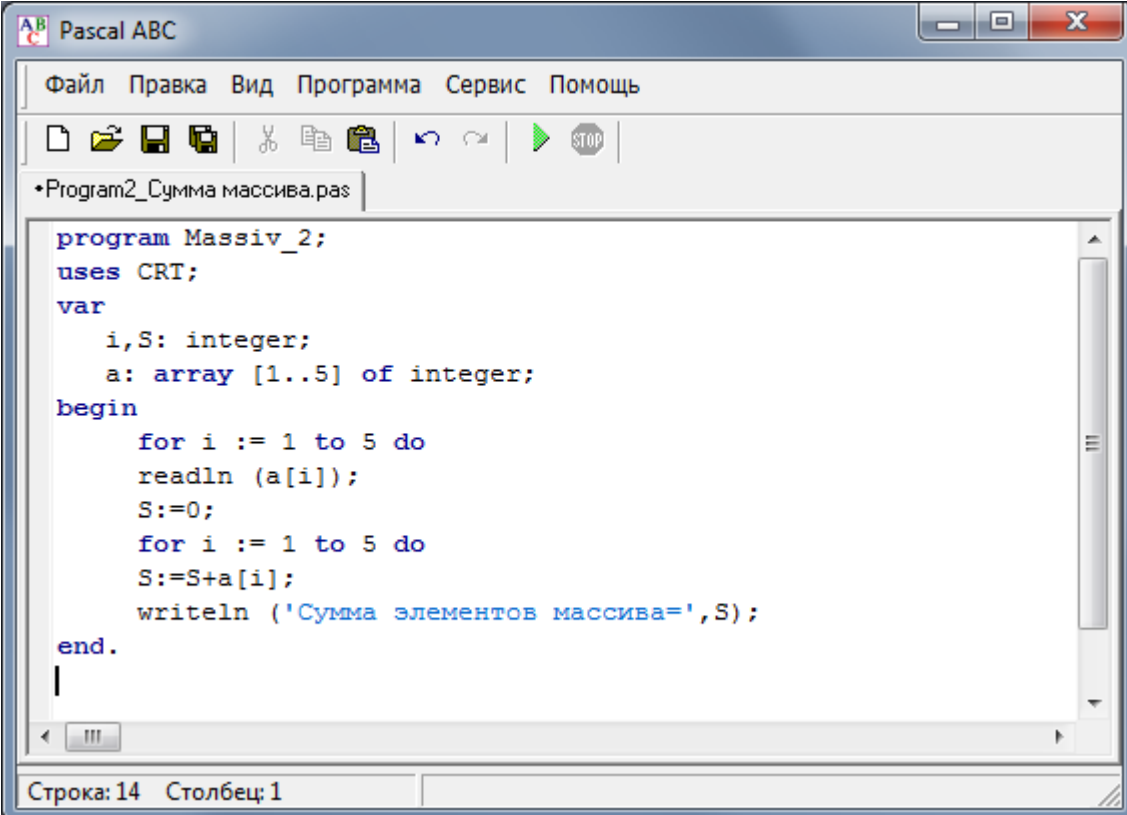
Тестовые входные данные:

массив **a**

1	2	3	4	5
---	---	---	---	---

Сумма элементов массива должна получиться равной 15 ($1+2+3+4+5=15$)

Текст программы:



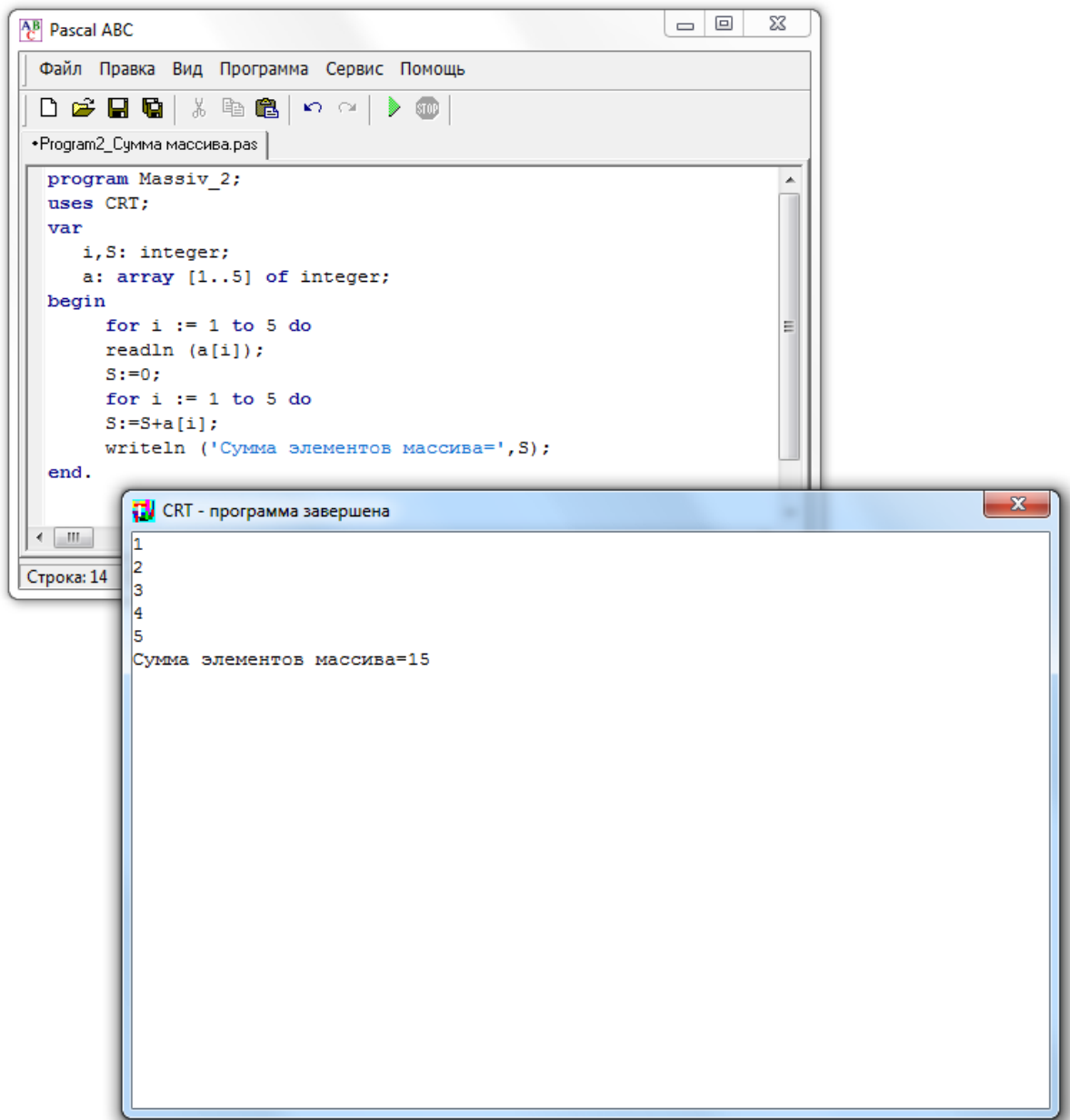
```
program Massiv_2;
uses CRT;
var
  i, S: integer;
  a: array [1..5] of integer;
begin
  for i := 1 to 5 do
    readln (a[i]);
  S:=0;
  for i := 1 to 5 do
    S:=S+a[i];
  writeln ('Сумма элементов массива=', S);
end.
```

Строка: 14 Столбец: 1

Мы ввели новую целочисленную величину – **S**, которая будет содержать результат суммирования.

Перед началом цикла, который на каждом шаге прибавляет к **S** значение очередного элемента массива, мы присвоили сумме значение 0 (Ноль).

Выполнение программы:



Для тестирования программы можно было использовать и более простые входные данные: например, ввести все элементы массива со значением **1** (Единица). Тогда сумма должна была получиться равной **5** (Пяти).

Пример 3. Нахождение элемента, отвечающего какому-либо условию, в одномерном целочисленном массиве.

Запишем программу, в которой осуществляется нахождение элемента массива, равного 0 (нулю) и его номера в целочисленном массиве **a**, состоящем из 5 элементов.

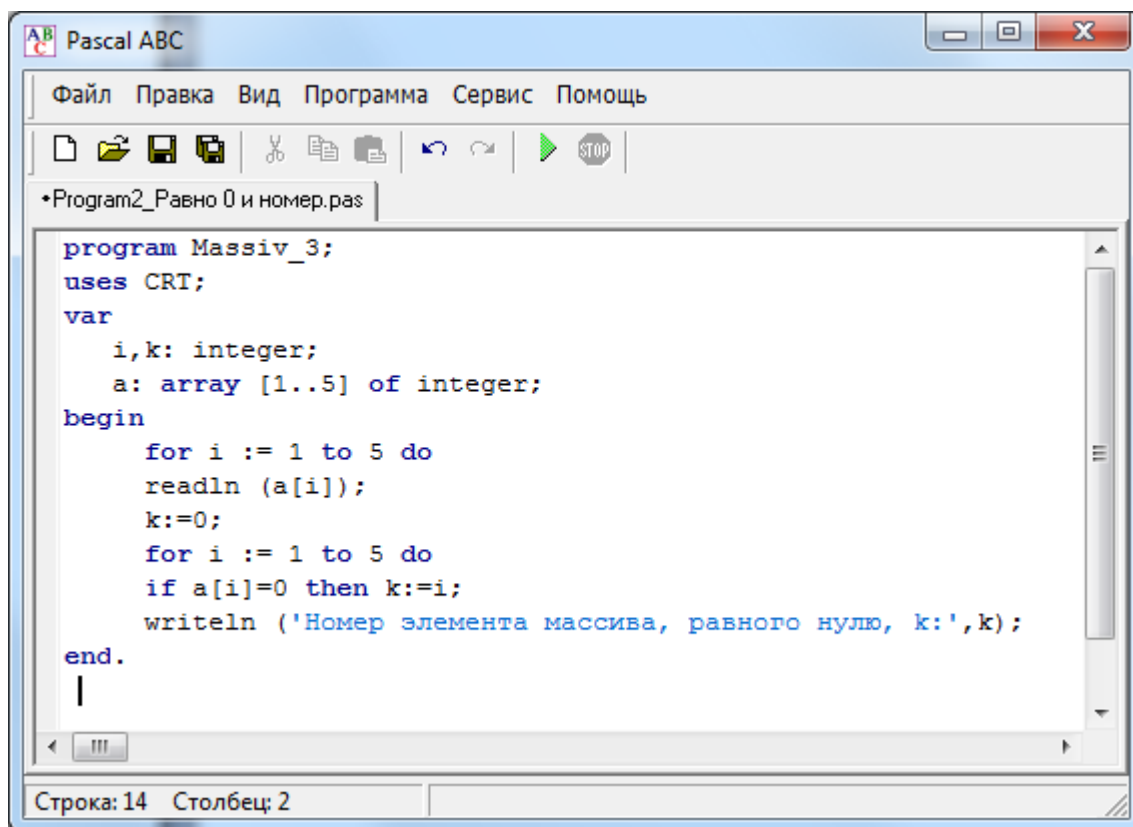
Тестовые входные данные:

*массив **a***

№	1	2	3	4	5
	1	5	3	0	2

Номер искомого элемента должен получиться равным 4.

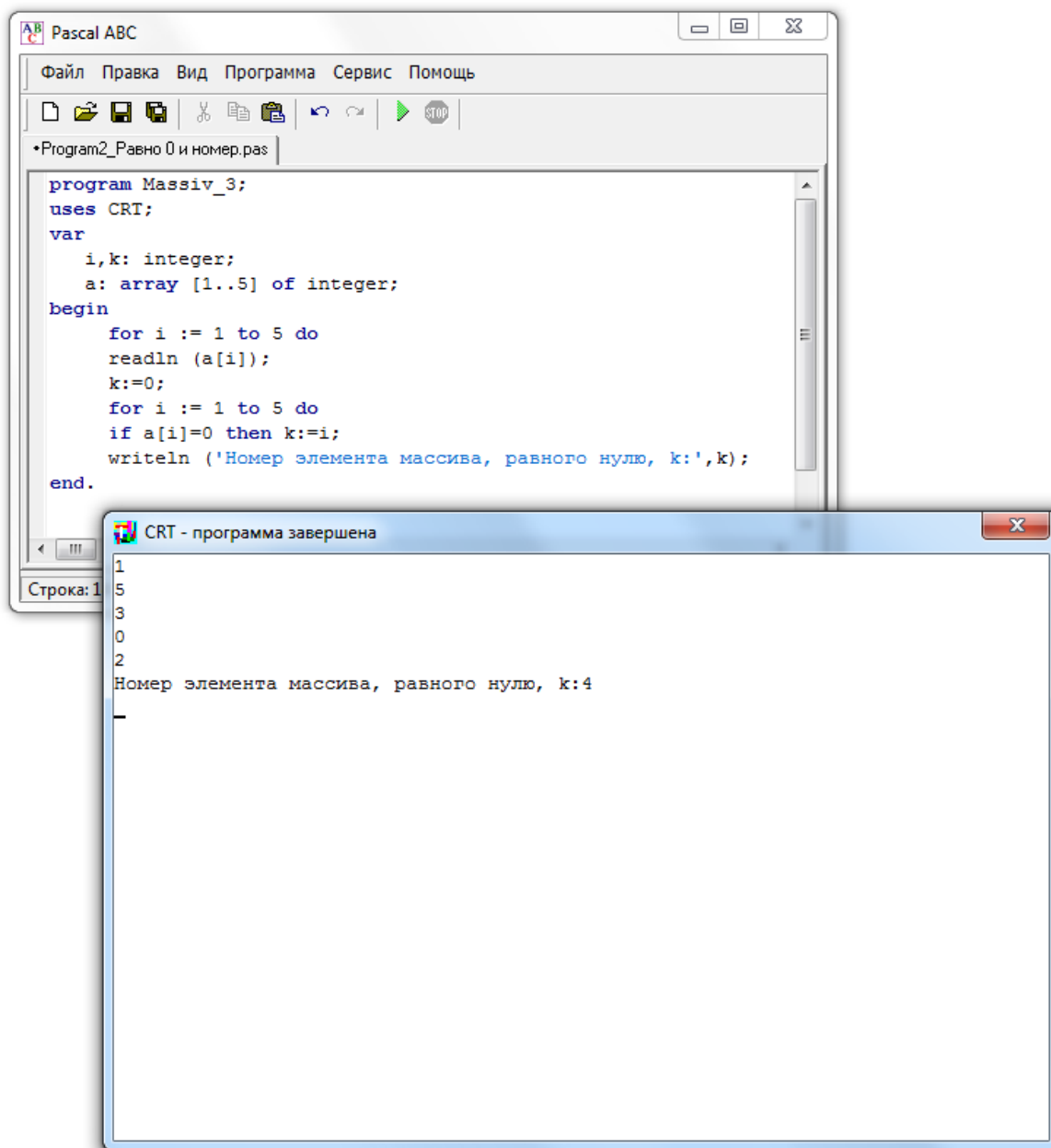
Текст программы:



```
program Massiv_3;
uses CRT;
var
  i,k: integer;
  a: array [1..5] of integer;
begin
  for i := 1 to 5 do
    readln (a[i]);
  k:=0;
  for i := 1 to 5 do
    if a[i]=0 then k:=i;
  writeln ('Номер элемента массива, равного нулю, k:',k);
end.
```

Здесь: *k* – номер искомого элемента, который записывается как значение данной переменной, если условие (элемент массива = 0) выполняется (имеет значение *истинно*).

Выполнение программы:



The image shows a screenshot of the Pascal ABC IDE and a CRT window. The IDE window, titled "Pascal ABC", displays the following code in a file named "Program2_Равно 0 и номер.pas":

```
program Massiv_3;
uses CRT;
var
  i,k: integer;
  a: array [1..5] of integer;
begin
  for i := 1 to 5 do
    readln (a[i]);
    k:=0;
    for i := 1 to 5 do
      if a[i]=0 then k:=i;
    writeln ('Номер элемента массива, равного нулю, k:',k);
end.
```

The CRT window, titled "CRT - программа завершена", shows the output of the program:

```
1
5
3
0
2
Номер элемента массива, равного нулю, k:4
```

Таким же образом, можно определить любое условие, которое будет проверяться внутри цикла обработки массива и запомнит результат в указанной переменной величине.

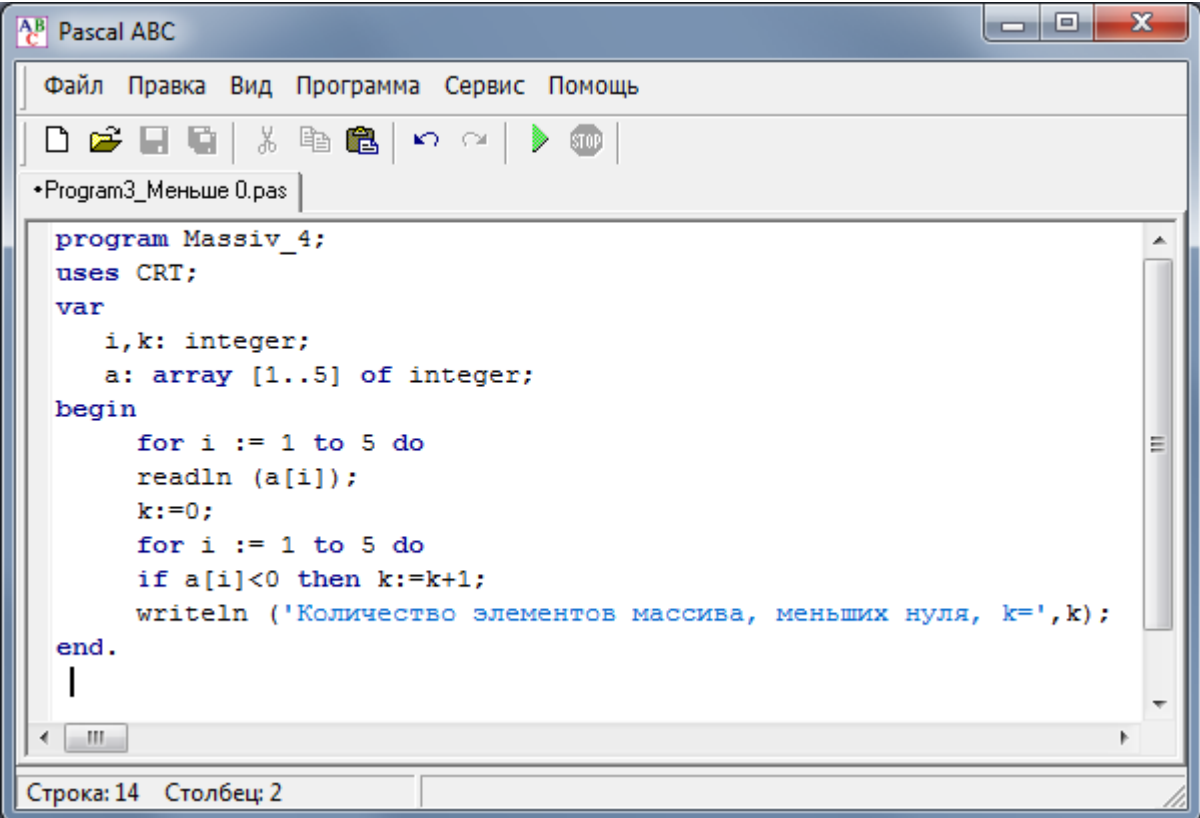
Например, в подобном массиве определим количество отрицательных элементов (т.е., имеющих значение меньше нуля).

Тестовые входные данные:
массив **a**

№	1	2	3	4	5
	2	-5	1	-4	-2

Количество искомых элементов должно получиться равным 3 (элементы №№ 2,4 и 5).

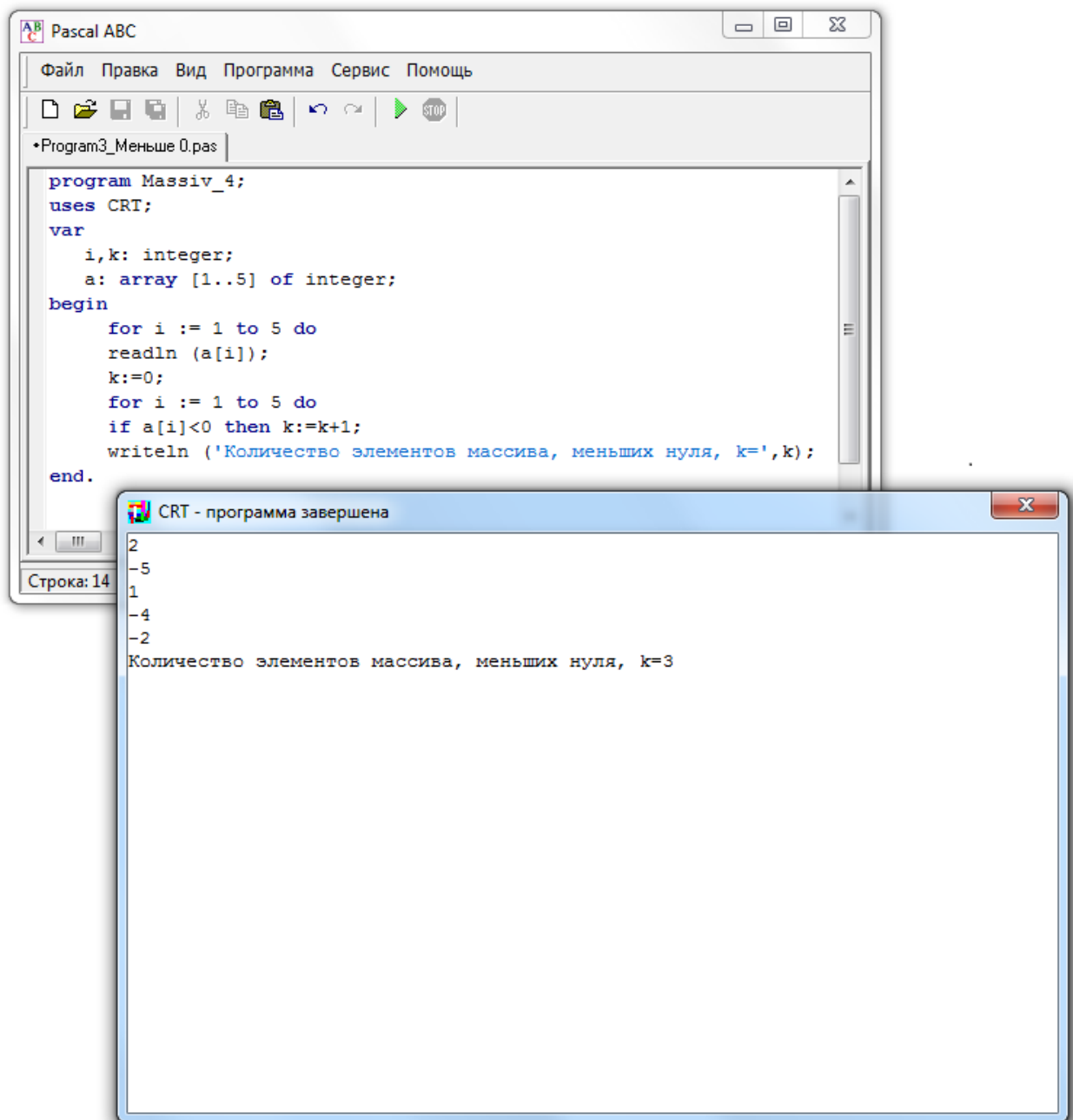
Текст программы:



```
program Massiv_4;
uses CRT;
var
  i,k: integer;
  a: array [1..5] of integer;
begin
  for i := 1 to 5 do
    readln (a[i]);
  k:=0;
  for i := 1 to 5 do
    if a[i]<0 then k:=k+1;
  writeln ('Количество элементов массива, меньших нуля, k=',k);
end.
```

Здесь: k – счётчик искомых элементов, который увеличивается на каждом шаге на 1 (Единицу), если условие (элемент массива <0) выполняется (имеет значение *истинно*).

Выполнение программы:



В качестве операций сравнения используют стандартные:

=	равно
>	больше
<	меньше
<>	не равно
>=	больше или равно
<=	меньше или равно

Пример 4. Нахождение максимального (наибольшего) или минимального (наименьшего) элемента массива в одномерном целочисленном массиве.

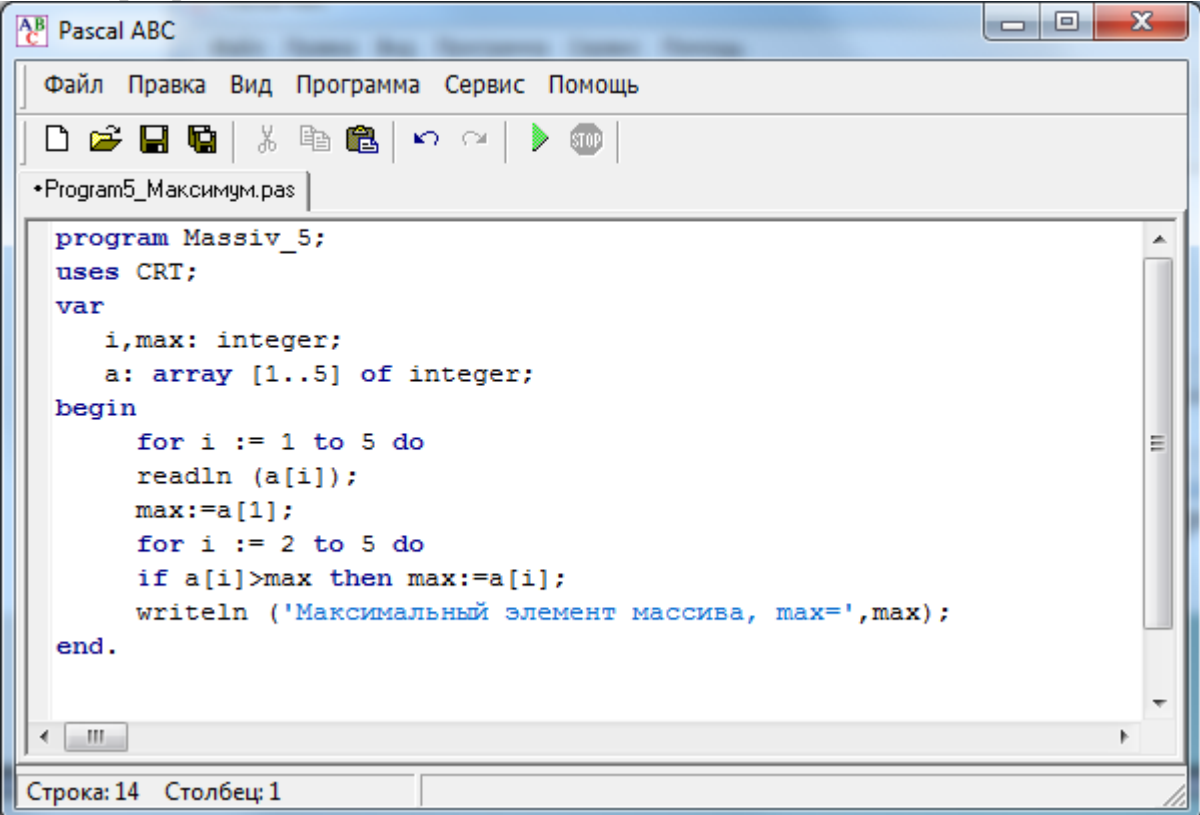
Запишем программу, в которой осуществляется нахождение максимального элемента массива в целочисленном массиве **a**, состоящем из 5 элементов.

Тестовые входные данные:
массив **a**

№	1	2	3	4	5
	4	7	21	12	18

Значение максимального элемента массива должен получиться равным 21.

Текст программы:

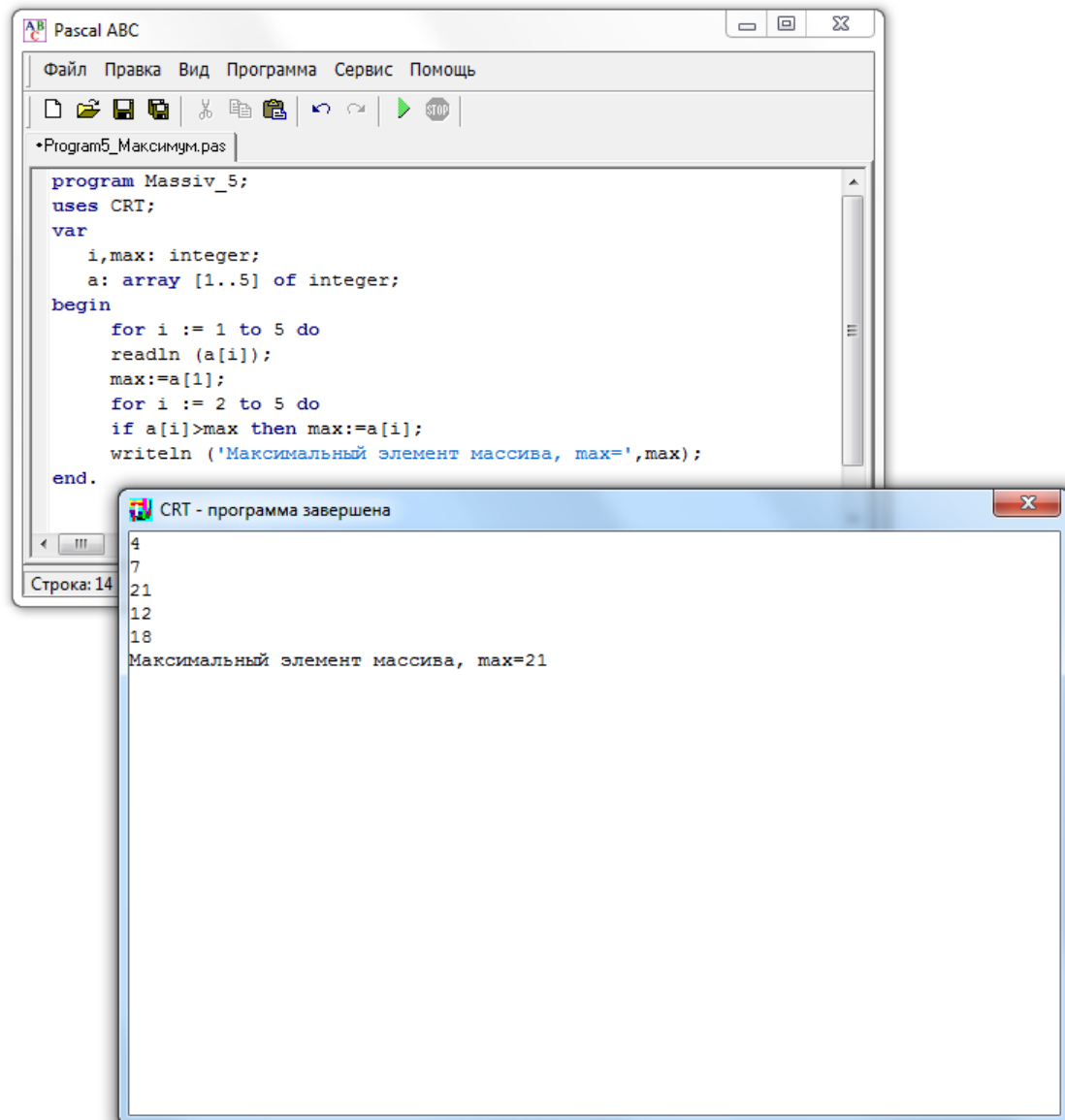


```
program Massiv_5;
uses CRT;
var
  i,max: integer;
  a: array [1..5] of integer;
begin
  for i := 1 to 5 do
    readln (a[i]);
  max:=a[1];
  for i := 2 to 5 do
    if a[i]>max then max:=a[i];
  writeln ('Максимальный элемент массива, max=',max);
end.
```

Строка: 14 Столбец: 1

Здесь: **max** – имя искомой величины (наибольшего элемента). Перед началом поиска ей присваивается значение первого элемента массива. Затем, в цикле, начиная со 2-го элемента мы начинаем сравнение. Если очередной элемент оказывается больше **max**, последнему присваивается значение этого элемента. Если условие не выполняется – никаких действий не производится, происходит переход к проверке следующего элемента массива.

Выполнение программы:



The screenshot shows the Pascal ABC IDE with a file named 'Program5_Максимум.pas'. The code defines a program 'Massiv_5' that uses the 'CRT' unit. It declares variables 'i, max' as integers and 'a' as an array of 5 integers. The program reads 5 integers from the user, then finds the maximum value by comparing each element starting from the second one. Finally, it prints the maximum value. A separate CRT window titled 'CRT - программа завершена' displays the input sequence: 4, 7, 21, 12, 18, followed by the output: 'Максимальный элемент массива, max=21'.

```
program Massiv_5;
uses CRT;
var
  i,max: integer;
  a: array [1..5] of integer;
begin
  for i := 1 to 5 do
    readln (a[i]);
  max:=a[1];
  for i := 2 to 5 do
    if a[i]>max then max:=a[i];
  writeln ('Максимальный элемент массива, max=',max);
end.
```

Строка: 14

CRT - программа завершена

4
7
21
12
18
Максимальный элемент массива, max=21

Для нахождения минимального элемента, достаточно в цикле поменять в условии проверки знак «больше» на «меньше».

Так же, очевидно, что если нужно запомнить номер искомого элемента массива, нужно в цикл ввести оператор присваивания текущего номера элемента, какой-либо величине.

По окончании всего цикла, в этой величине будет содержаться номер искомого элемента.

Литература:

- 1) **Информатика** : учебник для 9 класса / Босова Л.Л., Босова А.Ю. — М. : БИНОМ. Лаборатория знаний, 2013. — 184 с. : ил. - § 2.2, вопросы и задания (стр. 64-75).
- 2) Босова Л. Л. **Информатика : рабочая тетрадь** для 9 класса : в 2 ч. Ч 1. / Л. Л. Босова, А. Ю. Босова. — М. : БИНОМ. Лаборатория знаний, 2016. — 96 с.: ил. – Задания №№ 68-83 (стр. 77-96).

Программное обеспечение:

Среда программирования на языке Паскаль – Pascal ABC (версия 3.0.1)

Задания на практическую работу:

- 1) суммирование элементов массива;
- 2) поиск элемента с заданными свойствами (>0 , <0 , $=0$ и т.п.);
- 3) поиск максимального элемента массива;
- 4) поиск минимального элемента массива;
- 5) поиск номера максимального элемента массива;
- 6) поиск номера минимального элемента массива;
- 7) поиск количества элементов с заданными свойствами.