

Міністерство освіти і науки України
Харківський національний автомобільно-дорожній університет
Кафедра інформаційних технологій та мехатроніки

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторних робіт з дисципліни
“Інформаційні технології”

**“Програмування на мові С++
у середовищі Microsoft Visual Studio 2010”**

для студентів напряму підготовки 6.050702 ” Електромеханіка ”,
галузь знань 0507 ” Електротехніка та електромеханіка ”

Розроблено та надруковано доц. Симбірським Г.Д.

Харків, 2018

Лабораторная работа № 5 ИССЛЕДОВАНИЕ ОПЕРАЦИЙ С ОДНОМЕРНЫМИ МАССИВАМИ ДАННЫХ В VISUAL C++ 2010

Цель работы: получение навыков разработки программ для одномерных массивов данных в Visual C++ 2010.

1. Одномерные массивы данных и их инициализация в Visual C++ 2010

Массив – это конечная именованная последовательность однотипных величин.

Массив в информатике – это некоторое множество мест в памяти компьютера, называемых **элементами массива**, к которым можно обратиться по одному имени переменной. Каждый из **элементов** хранит единицу данных определенного типа (тип данных одинаков для всех элементов массива).

Каждый элемент массива определяется именем массива и его порядковым номером в массиве (индексом). Индекс элемента массива – всегда целое число.

Массивы бывают одномерными и многомерными. В данной работе исследуем работу с **одномерными массивами в Visual C++ 2010**. Одномерные массивы еще называют **векторами**.

Для использования массива в программе его необходимо **объявить**, т. е. зарезервировать под массив определенное количество ячеек памяти.

Гггггг

При объявлении массива указывается тип элементов массива, имя массива и его размер:

Тип Имя массива[размер]; .

Например, оператор

int A[8];

описывает целый одномерный массив по имени **A** из 8-и целых чисел. В памяти будет зарезервировано место для 8-и целочисленных элементов массива (таб. 5.1).

Таблица 5.1. Значения и расположение в памяти элементов одномерного массива **A[8]**

| Элемент массива | A[0] | A[1] | A[2] | A[3] | A[4] | A[5] | A[6] | A[7] |
|--------------------|------|------|------|------|------|------|------|------|
| Индекс элемента, i | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Значение элемента | 15 | 22 | 34 | 57 | 11 | 29 | 89 | 47 |

Обращение к элементам массива осуществляется по имени массива с указанием индекса (номера элемента массива) в квадратных скобках:

A[7]= 47;

x = A[3];

v = A[5];

Если объявлен массив

int B[100]; ,

то элементы массива будут иметь следующие индексы:

B[0], B[1], ... B[99].

Тогда оператор

x=B[13];

означает, что переменной **x** будет присвоено значение 14-го элемента массива **B**.

Массив, как и переменную, можно инициализировать при объявлении. Значения для последовательных элементов массива отделяются один от другого запятыми и помещаются в фигурные скобки. Например,

int C[6]= {2, 4, 7, 11, 12, 13}; .

Если в списке инициализации значений элементов указано меньше, чем размер массива, то имеет место частичная инициализация. При таком объявлении в выражении инициализации после последнего значения для наглядности ставят запятую:

int C[6]= {2, 4, 7,}; .

При этом элементам **C[0]**, **C[1]** и **C[2]** будут присвоены значения 2, 4 и 7, а оставшиеся элементы массива инициализации не получат.

Кроме этого, инициализация массивов возможна в процессе выполнения программы – путем записи данных в отведенные для массивов ячейки памяти.

При работе с массивами, в т. ч. одномерными, целесообразно использовать оператор цикла **for**, т.к. известен размер обрабатываемого массива (число элементов массива), т. е. число повторений цикла.

В языке C++ не проверяется выход индекса за пределы массива. Если массив **m[100]** целочисленный массив:

int m[100]; ,

а в программе указано

x=m[200]; ,

то сообщение об ошибке не будет, а переменной **x** будет присвоено произвольное значение.

При обработке массивов в Visual C++ 2010 все действия в программе выполняются над элементами массива (!), а не над массивом в целом. При этом индекс элемента может быть задан либо его значением, либо выражением:

A[4], F[i+k+1]; .

Над массивами можно выполнять следующие действия:

1. Вводить массивы в память компьютера.
2. Выводить массивы на экран дисплея, на другое устройство или в файл.
3. присваивать определенные значения элементам массивов.
4. Копировать массивы.
5. Переставлять элементы массивов.
6. Сортировать элементы массивов.

2. Консольный ввод и вывод одномерных массивов в среде Visual C++ 2010

Т. к. все действия необходимо выполнять над элементами массива, то для ввода массива в память компьютера необходимо организовать его поэлементный ввод посредством оператора цикла **for** (т. к. известен размер массива).

Задание 5.1. Исследовать способы консольного ввода и вывода массива **A** из **N** целых чисел. Необходимо ввести с клавиатуры массив **A** в память, определить его размер и вывести эту информацию на дисплей.

Решение. Блок-схема алгоритма решения такой задачи приведена на рис. 5.1:

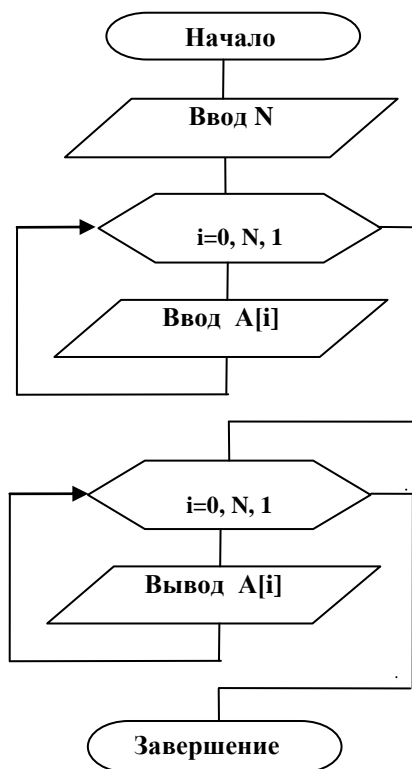


Рис. 5.1. Блок-схема алгоритма ввода и вывода элементов вектора с использованием цикла **for**

Реализующая данный алгоритм программа имеет следующий вид:

```
#include "stdafx.h"
#include <conio.h>
#include "iostream"
using namespace std;

int _tmain(int argc, _TCHAR* argv[])
{
    int const N=8; //инициализация размерности массива
    int i; //объявление параметра цикла i
    int A[N]; //объявление одномерного массива A
    cout<<endl<<"Vvedite massiv:"<<endl;
    for(i=0; i<N; i++) //цикл по i для ввода массива A
        cin>>A[i]; //ввод i-го элемента массива A
    cout<<endl;
    for (i=0; i<N; i++) //цикл по i для вывода массива A на экран
        cout<<" A["<<i<<"]= "<<A[i]; //вывод i-го элемента массива A на экран
    cout<<endl<<"size= "<<i; //вывод размерности массива A на экран
    getch();
    return 0;
}
```

}

После запуска программы и ввода элементов массива **A** вид экрана будет следующий:

Vvedite massiv:

1 2 3 4 5 6 7 8

**A[0]= 1 A[1]= 2 A[2]= 3 A[3]= 4 A[4]= 5 A[5]= 6 A[6]= 7 A[7]= 8
size= 8**

Создайте в текстовом процессоре **Word** файл **Результат_Фамилия_Лр6**. Поля документа сделайте по 0,5 см.

Поместите окно DOS с результатами решения **Задания 5.1** в центральной части окна **Microsoft Visual Studio** ниже программного кода **Lr5-1.cpp** (см. рис. 1.2) и нажмите клавишу **<Prt Scr>**, после чего вставьте полученную копию экрана в файл **Результат_Фамилия_Лр5**. Над вставленным рисунком проставьте номер задания – **5-1**. Файл результатов не закрывайте до получения оценки за выполненную практическую часть работы в тетрадь с отчетом.

Закройте окно DOS, откройте пункт меню **Файл** и выполните команду **Закреть решение**.

!Внимание! Результаты следующих заданий данной лабораторной работы сохраняйте строго в соответствии с приведенным выше порядком действий!

3. Исследование операций с одномерными массивами в среде Visual C++ 2010

Элементам массива могут быть **присвоены** значения выражений. При этом элементы массива и значения выражений должны иметь один и тот же тип.

Например, объявлен массив

```
double A[3];
```

тогда возможна запись

```
A[0]= 3.5;  
A[1]= 0;  
A[2]= a*x + b;
```

Копирование – это присваивание значений элементов одного массива элементам другого массива. При копировании оба массива должны иметь одинаковый размер и тип элементов. Копирование массива **A** в **B** будет иметь вид:

```
for (i=0; i<N; i++) B[i]= A[i];
```

Задание 5.2. Задать массив **X** действительных чисел из 8 элементов непосредственным присваиванием в программном коде. Вычислить массив **Y** по заданной формуле:

$$Y_i = 2X_i + \sqrt[3]{X_i^5} .$$

Массив **Y** вывести на экран.

Решение. Запишем формулу для расчета элементов массива **Y** на языке C++:

```
Y [i]=2*X[i]+pow(X[i], 5/3).
```

Блок-схема алгоритма решения этой задачи аналогична блок-схеме, приведенной на рис. 5.1. После блока **Вывод A[i]** необходимо включить операционный блок для расчета **Y [i]=2*X[i]+pow(X[i], 5/3)**.

Реализующая данный алгоритм программа имеет следующий вид:

```
#include "stdafx.h"  
#include <conio.h>  
#include "iostream"  
using namespace std;  
int _tmain(int argc, _TCHAR* argv[])  
{  
int i;  
double X[8]={1.2, 3.4, 12, 5.12, 23.1, 3, 0, 35.2}; //Инициализация элементов массива X[8]  
double Y [8];  
for (i=0; i<8; i++) //Цикл для вычисления и вывода элемента Y[i]  
{ //Начало составного оператора  
Y [i]=2*X[i]+pow(X[i], 5/3); //Вычисление элементов массива Y[8]  
cout<<" X["<i<<"]="<<X[i]; //Вывод элементов массива X[8] на экран  
cout<<" Y["<i<<"]="<<Y[i]; //Вывод элементов массива Y[8] на экран  
cout<<endl; //Переход на новую строку  
}  
getch();  
return 0;  
}
```

В результате выполнения программы получим:

```
X[0]=1.2 Y[0]=3.6  
X[1]=3.4 Y[1]=10.2
```

$X[2]=12$ $Y[2]=36$
 $X[3]=5.12$ $Y[3]=15.36$
 $X[4]=23.1$ $Y[4]=69.3$
 $X[5]=3$ $Y[5]=9$
 $X[6]=0$ $Y[6]=0$
 $X[7]=35.2$ $Y[7]=105.6$

Задание 5.3. Самостоятельно разработать программу для выполнения следующих действий:

1. Задать массив X действительных чисел из 8 элементов непосредственным присваиванием в программном коде:
 $X[8]=\{3.9 \ 2.5 \ 3.6 \ 6.2 \ 5.0 \ 3.3 \ 2.7 \ 4.6\}$.

2. Вычислить массив Y по заданной формуле, согласно своему варианту (таблица 5.1).

!!! Определить свой вариант как номер компьютера.

3. Массивы X и Y вывести на экран.

Перед вводом программного кода самостоятельно разработать и начертить в отчете по лабораторной работе блок-схему алгоритма решения данной задачи и записать формулу для расчета на языке C++.

Таблица 5.1 Исходные данные и формулы для расчета Y (Задание 5.3)

| № варианта | Формула для расчета Y | Значение a |
|------------|---|--------------|
| 1 | $Y_i = \operatorname{tg} \frac{1-x_i}{1+x_i} + \sin^2 5x_i + e^{5a}$ | $a=1,88$ |
| 2 | $Y_i = \frac{\sin \frac{x_i+1}{4}}{\sin^2 5x_i + e^{3a}}$ | $a=0,56$ |
| 3 | $Y_i = \frac{\sin^3(x_i+a) - \cos^2(x_i+a)}{(x_i+a)^4}$ | $a=1,77$ |
| 4 | $Y_i = \frac{\operatorname{tg}^3(x_i+a) - \arccos^2(x_i+a)}{(x_i+a)^4}$ | $a=1,33$ |
| 5 | $Y_i = \operatorname{ctg} \frac{1-3x_i}{1+2x_i} + \cos^2 5x_i + e^{3a}$ | $a=0,46$ |
| 6 | $Y_i = \frac{\cos^3(x_i+a) - 7(x_i+a)}{\operatorname{tg}(x_i+a)^4}$ | $a=1,82$ |
| 7 | $Y_i = \frac{\cos\left(\frac{3a+x_i}{4}\right)}{\sin^3 3x_i + e^{4a}}$ | $a=0,82$ |
| 8 | $Y_i = \frac{\sin^3(x_i+a) - \cos^2(x_i+a)}{(x_i+a)^4}$ | $a=1,47$ |
| 9 | $Y_i = \frac{\operatorname{tg} \frac{4a^2+1}{4}}{\cos^3 2x_i + e^{2a}}$ | $a=1,34$ |
| 10 | $Y_i = \sin \frac{1-x_i}{1+x_i} + \operatorname{tg}^4 5x_i + e^{5a}$ | $a=1,28$ |
| 11 | $Y_i = \frac{\sin^3(x_i+a) - \arccos^2(x_i+a)}{\cos(x_i+a)^4}$ | $a=0,66$ |
| 12 | $Y_i = \frac{\operatorname{ctg}\left(\frac{x_i^3+1}{4}\right)}{\cos^2 5x_i + e^{3a}}$ | $a=1,12$ |

| | | |
|----|--|--------|
| 13 | $Y_i = \frac{\text{ctg}^3(3x_i + a) - \sin^2(x_i + 7a)}{(5x_i + a)^3}$ | a=1,82 |
| 14 | $Y_i = \frac{\arcsin^3 \frac{4x_i + 1}{4}}{\text{ctg}^2(3x_i + e^{3a})}$ | a=1,42 |
| 15 | $Y_i = \frac{\sin^3 \frac{3x_i + 1}{2}}{\text{tg}^2 5x_i + e^{3a}}$ | a=1,72 |
| 16 | $Y_i = \frac{\sin^3(x_i + a) - \cos^2(x_i + a)}{(x_i + a)^4}$ | a=1,19 |
| 17 | $Y_i = \frac{\text{arccctg} \frac{x_i^3 + 1}{4}}{\cos^2 5x_i + e^{3a}}$ | a=0,12 |
| 18 | $Y_i = \frac{\text{tg}^3(x_i + a) - 5(\sin x_i + a)}{\sin^3(x_i + a)^4}$ | a=2,12 |

Задание 5.4. Задан массив **A** из **N** произвольных чисел. Массив необходимо инициализировать непосредственно в программе. Необходимо вычислить сумму элементов массива **A**. На экран вывести исходный массив **A**, сумму элементов массива **A** и число его элементов.

Решение. В качестве исходного можно использовать массив **X[8]** из задания 5.2.

Определим типы и структуры данных, которые будут использоваться в программе.

N – размер массива. Зададим его константой, равной, например, 8;

S – переменная для накопления суммы элементов массива;

i – параметр цикла;

Блок-схема алгоритма решения этой задачи приведена на рис. 5.2:

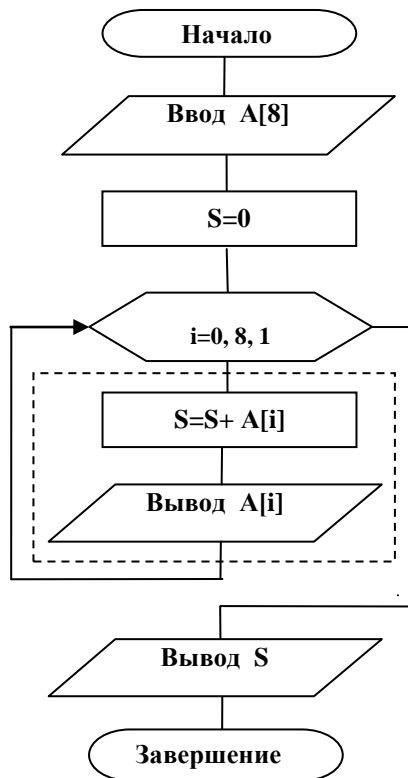


Рис. 5.2. Блок-схема алгоритма вычисления суммы элементов одномерного массива

Реализующая данный алгоритм программа имеет следующий вид:

```

#include "stdafx.h"
#include <conio.h>
  
```

```

#include "iostream"
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
int i;
double A[8]={1.2, 3.4, 12, 5.12, 23.1, 3, 0, 35.2}; //Инициализация массива A[8]
double S=0; //Начальное значение переменной S для суммирования
for (i=0; i<8; i++) //Цикл для вычисления суммы элементов массива A[8]
//и вывода его элементов на экран
{
S=S+A[i]; //Начало составного оператора
cout<<" A["<<i<<"]="<<A[i]; //Вычисление суммы элементов массива A[8]
//Вывод элементов массива A[8] на экран
} //Конец составного оператора
cout<<endl<<" S= "<<S<<" i= "<<i; //Вывод суммы и количества элементов массива A на экран
getch();
return 0;
}

```

В результате выполнения программы получим:

A[0]=1.2 A[1]=3.4 A[2]=12 A[3]=5.12 A[4]=23.1 A[5]=3 A[6]=0 A[7]=35.2
S=83.02 i=8

Задание 5.5. Самостоятельно разработать программу для выполнения следующих действий:

1. Задать массив **X** действительных чисел из 8 элементов непосредственным присваиванием в коде:

X[8]={3.9 2.5 3.6 6.2 5.0 3.3 2.7 4.6}.

2. Вычислить массив **Y** по заданной формуле, согласно своему варианту (таблица 5.1).

3. Вычислить сумму элементов массива **Y**.

Массив **X** необходимо инициализировать непосредственно в программном коде. На экран вывести исходный массив **X**, массив **Y**, сумму элементов массива **Y** и число его элементов.

Перед вводом программного кода самостоятельно разработать и начертить в отчете по лабораторной работе блок-схему алгоритма решения данной задачи и записать формулу для расчета на языке C++.

Задание 5.6. Составить программу ввода массива **X** произвольной длины от 1 до 20 элементов ($0 < k < 20$). Вычислить массив **Y** по следующей формуле:

$$y_i = \begin{cases} x_i^2 + \sin^3 3x_i, & \text{если } x > 4; \\ \sqrt{\ln x_i} - \operatorname{tg} x_i, & \text{если } x \leq 4. \end{cases}$$

Текущую длину массива $k=10$ и значения элементов массива $X(10)=1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ ввести с клавиатуры.

Решение. Блок-схема алгоритма решения этой задачи приведена на рис. 5.3.

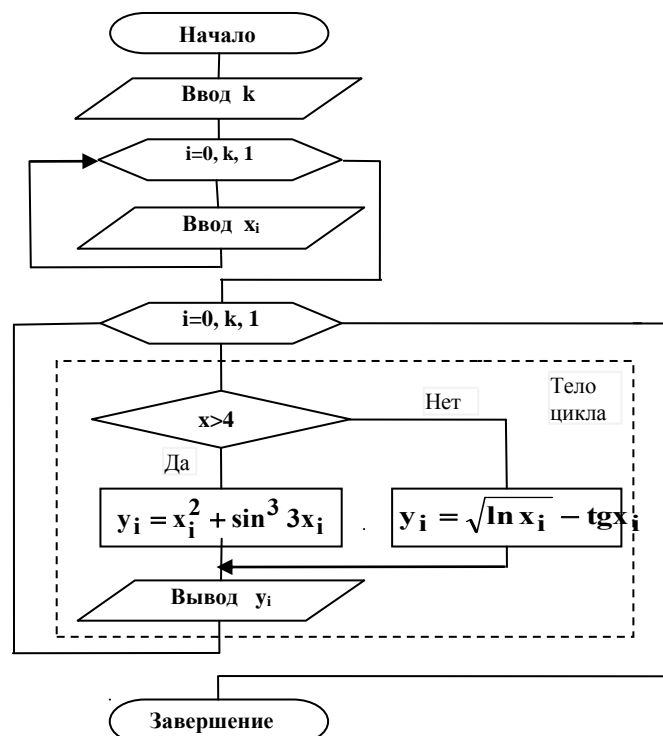


Рис. 5.3. Блок-схема алгоритма вычисления элементов одномерного массива в задании 5.6

Запишем формулы для вычисления массива Y на языке C++:

$$Y[i]=X[i]*X[i]+\text{pow}(\sin(3*X[i]),3) \text{ для } X[i]>4 \text{ и}$$

$$Y[i]=\sqrt{\log(X[i])}-\tan(X[i]) \text{ для } X[i]\leq 4.$$

Тогда программа имеет вид:

```
#include "stdafx.h"
#include <conio.h>
#include "iostream"
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
const int imax=20; //Максимальный размер массива (задается с запасом)
int i, k; //Счетчик и текущая длина массива X
float X[imax], Y[imax]; //Исходный массив X и входной массив Y
cout<<"Vvedite razmer massiva ot 1 do 20: ";
cin>>k; //Ввод длины массива X
for (i=0; i<k; i++) //Цикл для ввода элементов массива X
{ //Начало составного оператора
cout <<" Vvedite " <<i<<" element: ";
cin>>X[i]; //Ввод элементов массива X
} //Конец составного оператора
for (i=0; i<k; i++) //Цикл для вычисления массива Y
{ //Начало условного оператора
if(X[i]>4) //Вычисление массива Y для X[i]>4
Y[i]=X[i]*X[i]+pow(sin(3*X[i]),3);
else //Вычисление массива Y для X[i]<=0
Y[i]=sqrt(log(X[i]))-tan(X[i]);
cout<<" X["<<i<<"]="<<X[i]; //Вывод элементов массива X на экран
cout<<" Y["<<i<<"]="<<Y[i]<<endl; //Вывод элементов массива Y на экран
} //Конец составного оператора
getch();
return 0;
}
```

В результате выполнения программы получим:

Vvedite razmer massiva ot 1 do 20: 8

Vvedite 0 element: 1

Vvedite 1 element: 2

Vvedite 2 element: 3

Vvedite 3 element: 4

Vvedite 4 element: 5

Vvedite 5 element: 6

Vvedite 6 element: 7

Vvedite 7 element: 8

X[0]=1 Y[0]=-1.55741

X[1]=2 Y[1]=3.01759

X[2]=3 Y[2]=1.19069

X[3]=4 Y[3]=0.0195887

X[4]=5 Y[4]=25.275

X[5]=6 Y[5]=35.5765

X[6]=7 Y[6]=49.5857

X[7]=8 Y[7]=63.2574

Задание 5.7. Самостоятельно разработать программу для ввода массива X переменной длины от 1 до 20 элементов ($0 < k < 20$). Текущую длину массива $k=10$ и элементы массива X (1, 2, 3, 4, 5, 6, 7, 8, 9, 10) ввести с клавиатуры. Массив Y рассчитать по формуле в соответствии со своим вариантом (таб. 5.2). Определите свой номер варианта как номер компьютера.

Перед вводом программного кода самостоятельно разработать и начертить в отчете по лабораторной работе блок-схему алгоритма решения данной задачи и записать формулы для расчета на языке C++.

Таблица 5.2. Исходные данные и формулы для расчета Y (Задание 5.7)

| № вар. | $Y=f(x)$ | № вар. | $Y=f(x)$ |
|--------|--|--------|---|
| 1 | $y_i = \begin{cases} 5x_i^2 + \ln x_i, & \text{если } x < 0; \\ \lg 24^4 + 3x_i^6, & \text{если } x \geq 0; \end{cases}$ | 10 | $y_i = \begin{cases} 23.4^2 + 2\sqrt[3]{12.1x_i}, & \text{если } x < 1; \\ \text{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ |

| | | | |
|---|--|----|--|
| 2 | $y_i = \begin{cases} 23.4^2 + 2\sqrt[3]{12.1x_i}, & \text{если } x < 1; \\ \text{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ | 11 | $y_i = \begin{cases} \cos^2 x_i + 2\sin x_i, & \text{если } x < 1; \\ 2.5^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ |
| 3 | $y_i = \begin{cases} \cos^2 x_i + 2\sin x_i, & \text{если } x < 1; \\ 2.5^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ | 12 | $y_i = \begin{cases} \log_3 5x_i^2 + \sqrt{x_i^7}, & \text{если } x < 0; \\ e^{2x_i} + 3x_i^6, & \text{если } x \geq 0; \end{cases}$ |
| 4 | $y_i = \begin{cases} \log_3 5x_i^2 + \sqrt{x_i^7}, & \text{если } x < 0; \\ e^{2x_i} + 3x_i^6, & \text{если } x \geq 0; \end{cases}$ | 13 | $y_i = \begin{cases} 7.56 + \sin^2 x_i, & \text{если } x < 1; \\ 1.09^3 - x_i, & \text{если } x \geq 1; \end{cases}$ |
| 5 | $y_i = \begin{cases} 7.56 + \sin^2 x_i, & \text{если } x < 1; \\ 1.09^3 - x_i, & \text{если } x \geq 1; \end{cases}$ | 14 | $y_i = \begin{cases} 1.2^2 + 2\sqrt[3]{3x_i}, & \text{если } x < 1; \\ \text{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ |
| 6 | $y_i = \begin{cases} 1.2^2 + 2\sqrt[3]{3x_i}, & \text{если } x < 1; \\ \text{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ | 15 | $y_i = \begin{cases} \cos^2 x_i + 2\sin x_i, & \text{если } x < 1; \\ 3.2^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ |
| 7 | $y_i = \begin{cases} \cos^2 x_i + 2\sin x_i, & \text{если } x < 1; \\ 3.2^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ | 16 | $y_i = \begin{cases} 2x_i^3 + 4\sin x_i, & \text{если } x < 0 \\ \text{tg}^2 x_i^3, & \text{если } x \geq 0 \end{cases}$ |
| 8 | $y_i = \begin{cases} 2x_i^3 + 4\sin x_i, & \text{если } x < 0; \\ \text{tg}^2 x_i^3, & \text{если } x \geq 0; \end{cases}$ | 17 | $y_i = \begin{cases} 2\cos x_i^3 + 4\sin x_i, & \text{если } x < 0; \\ x_i^6, & \text{если } x \geq 0; \end{cases}$ |
| 9 | $y_i = \begin{cases} 2\cos x_i^3 + 4\sin x_i, & \text{если } x < 0; \\ x_i^6, & \text{если } x \geq 0; \end{cases}$ | 18 | $y_i = \begin{cases} 5x_i^2 + \ln x_i, & \text{если } x < 0; \\ 6.8^4 - \sqrt{x_i}, & \text{если } x \geq 0; \end{cases}$ |

4. Поиск элемента в одномерных массивах в среде Visual C++ 2010

Поиск элемента в массиве заключается в выделении из массива отдельных его элементов. Поиск может проводиться по образцу или по правилу.

Поиск по образцу заключается в следующем. Задается значение некоторой переменной (образец) и все элементы массива (или часть элементов) сравниваются со значением этой переменной (образцом).

Поиск по правилу проводится на основе проверки некоторых условий, которым должны отвечать либо элемент массива, либо группа элементов.

Задание 5.8. Исследуем программу для поиска элемента одномерного массива по образцу.

Задан массив **A** из 8 произвольных чисел. Необходимо определить количество элементов, которые больше заданного числа **q** и их порядковые номера. На экран вывести исходный массив, элементы, большие **q** и их порядковые номера.

Решение. Определим типы и структуры данных, которые будут использоваться в программе:

q – число, по которому осуществляется поиск элементов исходного массива **A**;

i – номер элемента исходного массива **A** (параметр цикла);

A[8] – массив из **N** произвольных чисел;

X[8] – массив для хранения номеров элементов исходного массива **A**, больших **q**.

j – номер элемента массива **X** для хранения номеров (параметр цикла);

Блок-схема алгоритма решения задачи поиска элементов одномерного массива приведена на рис. 5.4.

Тогда программа имеет вид:

```
#include "stdafx.h"
#include <conio.h>
#include "iostream"
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
    int i, j, k;
    double q=3.7; //Объявление переменной q
    double A[8]={1, 2, 3, 4, 5, 6, 7, 8}; //Инициализация элементов исходного массива A[8]
    int X[8]; //Объявление массива X[k] номеров элементов массива A[8]
    k=0;
    for (i=0; i<8; i++) // Цикл для перебора элементов массива A[8]
    { //Начало составного оператора в цикле
        cout<<" A["<<i<<"] = "<<A[i]; //Вывод всех элементов массива A[8]
```

```

if(A[i]>q)
{
X[k]=i;
k=k+1;
cout<<" >q  ";
}
}
cout<<endl<<"Nomera elementov > q: ";
for (j=0;j<k; j++)
cout<<X[j]<<" ";
getch();
return 0;
}

```

//Условный оператор для сравнения элементов массива A[8] с q
//Начало составного оператора в условном операторе
//Записываются номера элементов, больших q, в массив X[k]

//Вывод знака >q, если элемент массива A[8] больше, чем q
//Конец составного оператора в условном операторе
//Конец составного оператора в цикле

//Цикл для вывода номеров элем-в массива A[8], больших q
//Вывод номеров элем-в массива A[8], больших q

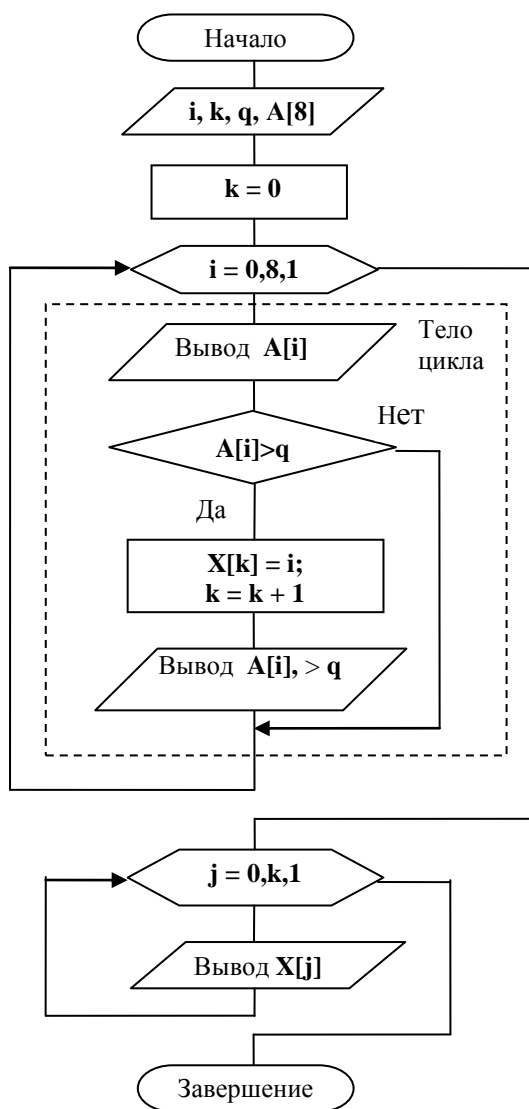


Рис. 5.4. Блок-схема алгоритма поиска элементов одномерного массива по образцу

В результате выполнения программы получим:

A[0] = 1 A[1] = 2 A[2] = 3 A[3] = 4 >q A[4] = 5 >q A[5] = 6 >q A[6] = 7 >q A[7] = 8 >q
Nomera elementov > q: 3 4 5 6 7

Задание 5.9. Исследуем программу для поиска элемента одномерного массива по правилу.

Задан массив A из 8 произвольных чисел. Найти максимальный элемент массива A и его номер. На экран вывести исходный массив A, максимальный элемент массива A и его номер.

Решение. Определим типы и структуры данных, которые будут использоваться в программе:

i – номер элемента исходного массива A (параметр цикла);

A[8] – массив из 8 произвольных чисел;

B – вспомогательная переменная для хранения максимального элемента массива для i-го шага;

C – вспомогательная переменная для хранения номера (индекса) максимального элемента массива.

Блок-схема алгоритма поиска максимального элемента одномерного массива приведена на рис. 5.5.

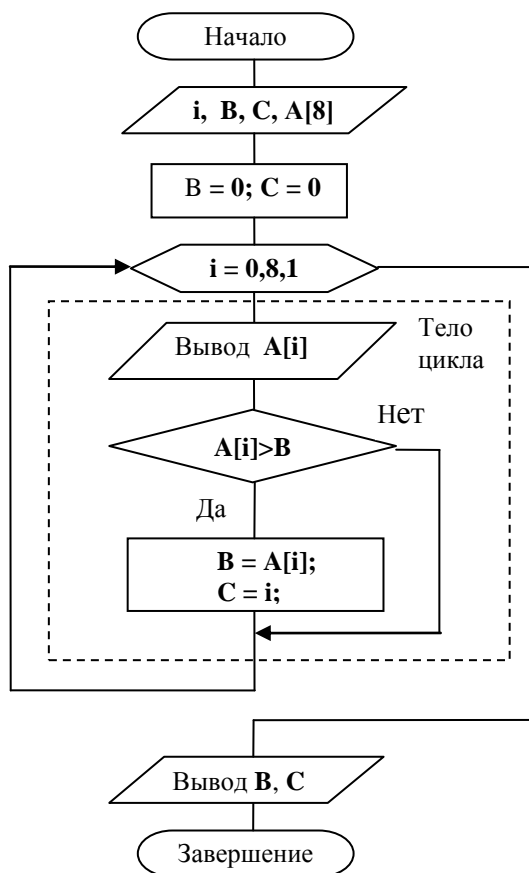


Рис. 5.5. Блок-схема алгоритма поиска максимального элемента одномерного массива

Программа поиска максимального элемента одномерного массива имеет вид:

```

#include "stdafx.h"
#include <conio.h>
#include "iostream"
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
int i, C; //Объявление вспомогательной переменной C и индекса i
double B; //Объявление вспомогательной переменной B
double A[8]={1, 2, 3, 4, 5, 6, 7, 1}; //Инициализация элементов исходного массива A[8]
B=A[0];
C=0;
for (i=0; i<8; i++) //Цикл для перебора элементов массива A[8]
{ //Начало составного оператора в цикле
cout<<" A["<<i<<"] = "<<A[i]; //Вывод всех элементов массива A[8]
if(A[i]>B) //Условный оператор для сравнения элементов массива A[8]
{ //Начало составного оператора в условном операторе
B=A[i]; //Присваивание переменной B значения максимального элемента
C=i; //Присваивание переменной C номера максимального элемента
} //Конец составного оператора в условном операторе
} //Конец составного оператора в цикле
cout<<endl<<"Max element: "<<B;
cout<<endl<<"N max elementa: "<<C;
getch();
return 0;
}
  
```

В результате выполнения программы получим:

```

A[0] = 1   A[1] = 2   A[2] = 3   A[3] = 4   A[4] = 5   A[5] = 6   A[6] = 7   A[7] = 1
Max element: 7
N max elementa: 6
  
```

Задание 5.10. Самостоятельно разработать программу для поиска максимального элемента одномерного массива Y . Массив Y рассчитать из исходного массива X по формуле в соответствии со своим вариантом (таб. 5.3). Массив X (1, 2, 3, 4, 5, 6, 7, 8, 9, 10) ввести в программном коде. Определите свой номер варианта как номер компьютера.

Перед вводом программного кода самостоятельно разработать и начертить в отчете по лабораторной работе блок-схему алгоритма решения данной задачи и записать формулы для расчета на языке C++.

Таблица 5.3. Исходные данные и формулы для расчета массива Y (Задание 5.10)

| № вар. | $Y=f(x)$ | № вар. | $Y=f(x)$ |
|--------|---|--------|---|
| 1 | $y_i = \begin{cases} \cos^2 x_i + 2 \sin x_i, & \text{если } x < 1; \\ 3 \cdot 2^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ | 10 | $y_i = \begin{cases} \cos^2 x_i + 2 \sin x_i, & \text{если } x < 1; \\ 3 \cdot 2^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ |
| 2 | $y_i = \begin{cases} 2x_i^3 + 4 \sin x_i, & \text{если } x < 0; \\ \operatorname{tg}^2 x_i^3, & \text{если } x \geq 0; \end{cases}$ | 11 | $y_i = \begin{cases} 2x_i^3 + 4 \sin x_i, & \text{если } x < 0 \\ \operatorname{tg}^2 x_i^3, & \text{если } x \geq 0 \end{cases}$ |
| 3 | $y_i = \begin{cases} 5x_i^2 + \ln x_i, & \text{если } x < 0; \\ \lg 24^4 + 3x_i^6, & \text{если } x \geq 0; \end{cases}$ | 12 | $y_i = \begin{cases} 23 \cdot 4^2 + 2\sqrt[3]{12 \cdot 1x_i}, & \text{если } x < 1; \\ \operatorname{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ |
| 4 | $y_i = \begin{cases} 23 \cdot 4^2 + 2\sqrt[3]{12 \cdot 1x_i}, & \text{если } x < 1; \\ \operatorname{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ | 13 | $y_i = \begin{cases} \cos^2 x_i + 2 \sin x_i, & \text{если } x < 1; \\ 2 \cdot 5^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ |
| 5 | $y_i = \begin{cases} \cos^2 x_i + 2 \sin x_i, & \text{если } x < 1; \\ 2 \cdot 5^3 - 3x_i, & \text{если } x \geq 1; \end{cases}$ | 14 | $y_i = \begin{cases} \log_3 5x_i^2 + \sqrt{x_i^7}, & \text{если } x < 0; \\ e^{2x_i} + 3x_i^6, & \text{если } x \geq 0; \end{cases}$ |
| 6 | $y_i = \begin{cases} \log_3 5x_i^2 + \sqrt{x_i^7}, & \text{если } x < 0; \\ e^{2x_i} + 3x_i^6, & \text{если } x \geq 0; \end{cases}$ | 15 | $y_i = \begin{cases} 7 \cdot 56 + \sin^2 x_i, & \text{если } x < 1; \\ 1 \cdot 09^3 - x_i, & \text{если } x \geq 1; \end{cases}$ |
| 7 | $y_i = \begin{cases} 7 \cdot 56 + \sin^2 x_i, & \text{если } x < 1; \\ 1 \cdot 09^3 - x_i, & \text{если } x \geq 1; \end{cases}$ | 16 | $y_i = \begin{cases} 1 \cdot 2^2 + 2\sqrt[3]{3x_i}, & \text{если } x < 1; \\ \operatorname{tg}^3 x_i, & \text{если } x \geq 1; \end{cases}$ |
| 8 | $y_i = \begin{cases} 2x_i^3 + 4 \sin x_i, & \text{если } x < 0; \\ \operatorname{tg}^2 x_i^3, & \text{если } x \geq 0; \end{cases}$ | 17 | $y_i = \begin{cases} 2 \cos x_i^3 + 4 \sin x_i, & \text{если } x < 0; \\ x_i^6, & \text{если } x \geq 0; \end{cases}$ |
| 9 | $y_i = \begin{cases} 2 \cos x_i^3 + 4 \sin x_i, & \text{если } x < 0; \\ x_i^6, & \text{если } x \geq 0; \end{cases}$ | 18 | $y_i = \begin{cases} 5x_i^2 + \ln x_i, & \text{если } x < 0; \\ 6 \cdot 8^4 - \sqrt{x_i}, & \text{если } x \geq 0; \end{cases}$ |

Контрольные вопросы

1. Приведите понятие массива.
2. Приведите понятие одномерного массива.
3. Перечислите основные свойства массивов.
4. Как объявляются одномерные массивы?
5. Как обращаются к элементам одномерных массивов?
6. Как и какие можно выполнять действия над одномерными массивами?
7. С помощью какого оператора рациональнее выполнять действия над одномерными массивами?
8. Каким оператором можно ввести с клавиатуры n элементов массива X ?
 - 1) `for (i=0; i<=n; i++) cin>>X[i]`
 - 2) `for (i=0; i<n; i++) cin>>X[i]`
 - 3) `for (i=1; i<=n; i++) cin>>X[i]`
9. Как в программе на C++ будет выведен на экран массив $A[k]$ оператором `for(i=0; i<k; i++) cout<<A[i]<<endl;`?
 - 1) В виде строки
 - 2) В виде столбца
 - 3) В виде матрицы
10. Какой оператор копирует массив $A[N]$ в массив $B[N]$?
 - 1) `for (i=0; i<N; i++) B[i]=A[i];`
 - 2) `for (i=0; i<K; i++) B[i]=A[i];`
 - 3) `for (i=0; i<N; i++) A[i]=B[i];`
11. Задан массив $X = \{X_1, X_2 \dots, X_N\}$. Каким оператором можно вывести этот массив на экран?
 - 1) `cout<<X;`
 - 2) `for (i=0; i<n; i++) cout<<X;`
 - 3) `for (i=0; i<n; i++) cout<<X[i];`