Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

# оТЧЕТ

по лабораторной работе

на тему:

Многомерные Массивы

|  |  |  |
| --- | --- | --- |
| Выполнил  Студент гр. 951002 |  | В. Н. Протасеня |
| Проверил |  | Асс. Е.Е. Фадеева |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск, 2019

1. Теоретические сведения по теме лабораторной работы

**Массив** – упорядоченная совокупность однотипных элементов, имеющих

общее имя. Данное имя называется полной переменной , так как ее значением

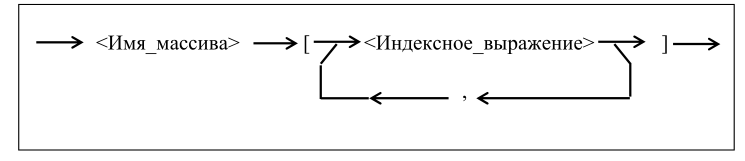
является весь массив.

**Тип элементов** массива называется базовым типом массива .

Для выделения отдельных элементов массива используются **индексы**

(индексные выражения). Индекс задает правило вычисления номера нужного

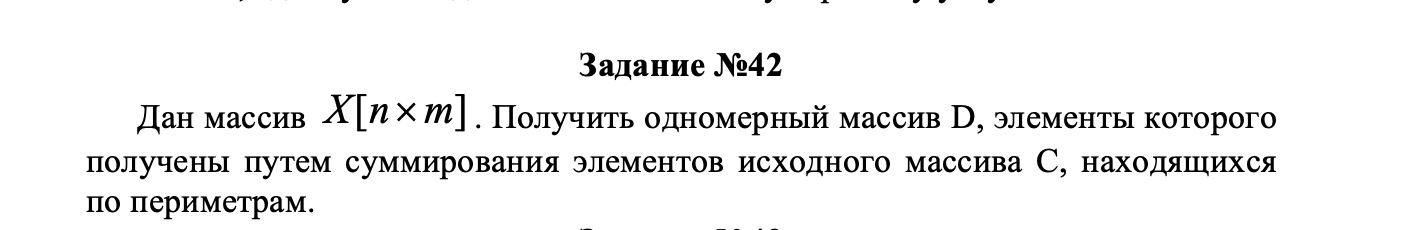
элемента массива.

****Для указания определенного элемента массива используется формат:

**Индексное выражение** – это выражение скалярного типа. Количество

индексных выражений определяется количеством измерений массива.

1.Задание на лабораторную работу



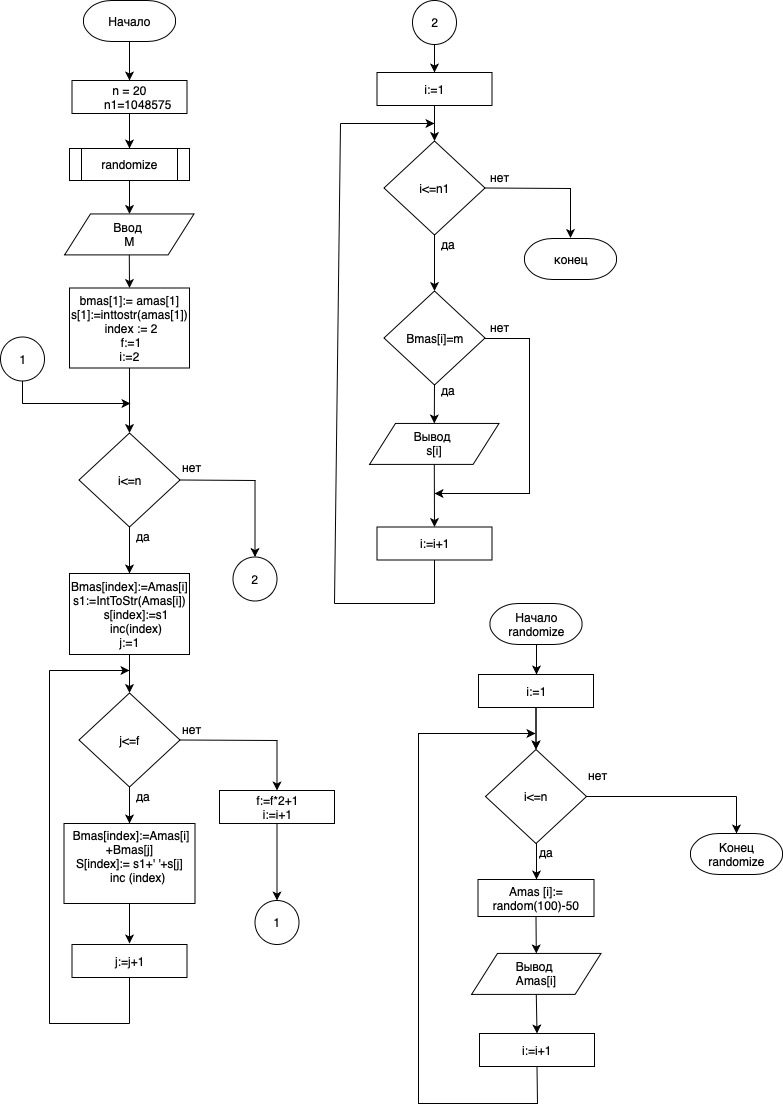
* 1. Эскиз ожидаемого результата

1. Выполнение
   1. Разработка алгоритма

Таблица 3.1 используемые идентификаторы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Имя идентификатора | назначение | Тип идентификатора | Начальное значение | Закон изменения | Имя цикла, в котором происходит изменение переменной |
| n | Размерность массива Amas | константа | 20 | — | — |
| n1 | Размерность массивов Bmas, S | константа | 1048575 | — | — |
| Amas | Начальный массив | Tmas = array [1 .. n] of integer | Ввод с клавиатуры | — | 35-74 |
| Bmas | Массив всех возможных сумм | Bmas1= array [1 .. n1] of integer | — | Bmas[index]:=Amas[i]+Bmas[j] | 50-74 |
| S | Массив для хранения элементов начального массива | StringMas= array [1 .. n1] of string | — | S[index]:= s[j]+' '+s1 | 51-74 |
| i | Параметр цикла | integer | 2 | i:=i+1 | 36-61 |
| index | Параметр цикла | integer | 2 | inc(index) | 36-61 |
| j | Параметр цикла | integer | 1 | j:=j+1 | 36-61 |
| f | Количество возможных сложений на данном шаге | integer | 1 | f:=f\*2+1 | 72 |
| S1 | Число, равное элементу начального массива | string |  | s1:= IntToStr(Amas[i]) | 56-70 |
| m | Число, которому равна сумма элементов начального массива | integer | Ввод с клавиатуры | — | 46-48 |

Рисунок 3.1 – Схема работы программы



Описание алгоритма: Программа состоит из трех циклов с предусловием с параметром и процедуры ввода randomize, которая включает в себя цикл с предусловием с параметром. Первый цикл – сложный, внешний, второй – простой, внутренний. Третий – простой.

3.2 Текст программы и его описание

program laba4;

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils, math;

const

n = 20;

n1=1048575; //2^n-1

// n=10 ;

// n1=1023;

//n=4;

//n1=15;

type

Tmas = array [1 .. n] of integer;

Bmas1= array [1 .. n1] of integer;

StringMas= array [1 .. n1] of string;

var

Amas: Tmas;

Bmas: Bmas1;

s: StringMas;

i, index,j,m,f: integer;

s1,s2: string;

begin

writeln('write initial massive:'); //ввод начального массива

//randomize;

for i:=1 to n do

begin

Amas [i]:= random(100)-50;

write(Amas[i], ' ');

end;

// for i := 1 to n do

// begin

// read(Amas[i]);

// end;

writeln;

writeln('write M:'); // ввод M

readln(m);

bmas[1]:= amas[1]; // присвоение начального значения bmass

s[1]:=inttostr (amas[1]);

index := 2;

f:=1;

for i := 2 to n do // заполение bmass

begin

Bmas[index] := Amas[i];

s1:= IntToStr(Amas[i]); //присвоение строкового массива

s[index] :=s1;

inc(index);

for j := 1 to f do

begin

Bmas[index]:=Amas[i]+Bmas[j];

S[index]:= s1+' '+s[j]; // строковый массив

inc (index);

end;

f:=f\*2+1;

end;

writeln;

writeln('result massives:');

for i:= 1 to n1 do

begin

//write(' ', Bmas[i]);

if (Bmas[i]=m) then

begin

writeln (S[i]);

end;

end;

readln;

end.

Описание программы: программа подсчитывает всевозможные вариации сумм элементов введенного (начального) массива и среди них находит суммы, равные введенному числу M.

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
| M>0 | 1 | 0 0 17 4 5 13 6 3 7 8  M= 15 | 6 5 4  6 5 4 0  6 5 4 0  6 5 4 0 0  7 3 5  7 3 5 0  7 3 5 0  7 3 5 0 0  8 3 4  8 3 4 0  8 3 4 0  8 3 4 0 0  8 7  8 7 0  8 7 0  8 7 0 0 | Тест пройден |
| M<0 | 2 | -50 -47 36 -30 -23 17 -19 -34 -13 -8  M= -21 | -34 -23 36  -8 17 -30  -8 -19 -30 36  -8 -19 17 36 -47  -8 -13  -8 -13 17 -23 -30 36  -8 -13 -34 -19 17 36 | Тест пройден |
| M=0 | 3 | -2 7 2 0 8 9 0 0 3 -5  M= 0 | 2 -2  0  0 2 -2  0  0 2 -2  0 0  0 0 2 -2  0  0 2 -2  0 0  0 0 2 -2  0 0  0 0 2 -2  0 0 0  0 0 0 2 -2  -5 7 -2  -5 0 7 -2  -5 0 7 -2  -5 0 0 7 -2  -5 0 7 -2  -5 0 0 7 -2  -5 0 0 7 -2  -5 0 0 0 7 -2  -5 3 2  -5 3 0 2  -5 3 0 2  -5 3 0 0 2  -5 3 0 2  -5 3 0 0 2  -5 3 0 0 2  -5 3 0 0 0 2 | Тест пройден |
| Массив состоит из нулей | 4 | 0 0 0 0  M=0 | 0  0  0 0  0  0 0  0 0  0 0 0  0  0 0  0 0  0 0 0  0 0  0 0 0  0 0 0  0 0 0 0 | Тест пройден |
| Массив состоит из больших чисел | 5 | 1509 1503 1504 1506  M=3007 | 1504 1503 | Тест пройден |

* 1. Анализ прохождения тестов и полученных результатов

Программа прошла тесты успешно и ошибок не обнаружено.

* 1. Итоговый текст программы

program laba5;

{$APPTYPE CONSOLE}

uses

SysUtils;

const

O = 2000;

type

Tmas = array [1 .. O] of Integer;

var

Amas, Bmas, Cmas: Tmas;

i, j, sravnenie, zamena, zamena1, sravnenie1, n, h: Integer;

s: string;

procedure Swap(var A, B: Integer); // меняет местами элементы массива

var

temp: Integer;

begin

temp := A;

A := B;

B := temp;

end;

procedure TableHead;

begin

Writeln('---------------------------------------------------------------------------------');

Writeln('| | Bubble Sort #1 | Insertion Sort #2 |');

Writeln('| Array |---------------|---------------|---------------|----------------|');

Writeln('| type | Number of | Number of | Number of | Number of |');

Writeln('| | comparisons | exchanges | comparisons | exchanges |');

Writeln('|---------------|---------------|---------------|---------------|----------------|');

end;

procedure Table;

begin

Writeln('|', n:5,

' | | | | |');

Writeln('| ', s, ' | ', sravnenie1:13, ' | ', zamena1:14, '|', sravnenie:15,

'| ', zamena:15, '|');

Writeln('|---------------|---------------|---------------|---------------|----------------|');

end;

procedure InsertionSort(var Amas: Tmas);

var

x: Integer;

begin

zamena := 0;

sravnenie := 0;

for i := 2 to n do

begin

x := Amas[i];

inc(zamena);

j := i - 1;

inc(sravnenie);

while (j >= 1) and (Amas[j] > x) do

begin

Amas[j + 1] := Amas[j];

inc(zamena);

j := j - 1;

inc(sravnenie);

end;

Amas[j + 1] := x;

inc(zamena);

end;

end;

procedure BubbleSort(var Amas: Tmas);

begin

zamena1 := 0;

sravnenie1 := 0;

for i := 1 to n - 1 do

begin

for j := 1 to n - i do

begin

if Amas[j] > Amas[j + 1] then

begin

Swap(Amas[j], Amas[j + 1]);

zamena1 := zamena1 + 3;

end;

inc(sravnenie1);

end;

end;

end;

procedure NewArray(var first, second: Tmas); // копирует аmas в bmas cmas

begin

for i := 1 to n do

begin

first[i] := Amas[i];

second[i] := Amas[i];

end;

end;

procedure reverse(var B: Tmas); // разворот массивов

var

tempmas: Tmas;

begin

j := 1;

for i := n downto 1 do

begin

tempmas[j] := B[i];

inc(j);

end;

for i := 1 to n do

begin

B[i] := tempmas[i];

end;

end;

begin

TableHead;

randomize;

for i := 1 to O do // заполнение массива начального

begin

Amas[i] := random(100) - 50;

end;

n := 10;

for h := 1 to 3 do

begin

NewArray(Bmas, Cmas); // создание массивов amas bmas

InsertionSort(Bmas); // сортировки выбором для несортированного

BubbleSort(Cmas); // сортировка пузырьком

s := 'el. unsorted ';

Table;

InsertionSort(Bmas);

BubbleSort(Cmas);

s := 'el. sorted '; // для отсортированного

Table;

reverse(Bmas); // разворо массивов

reverse(Cmas);

InsertionSort(Bmas); // сортировка для обратно отсортированных массивов

BubbleSort(Cmas);

s := 'el. bk sorted';

Table;

if n = 100 then

n := 2000;

if n = 10 then

n := 100;

end;

readln;

end.