**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа природных ресурсов

Направление подготовки Химическая технология

Отделение химической инженерии

**СОСТАВЛЕНИЕ ПРОГРАММ С ИСПОЛЬЗОВАНИЕМ ДВУМЕРНЫХ МАССИВОВ. ФАЙЛЫ**

**Лабораторная работа по дисциплине «Углубленный курс информатики»**

Выполнил студент гр. 2Д91 А.А. Циттель

(Подпись)

\_\_18\_\_\_ \_\_марта\_\_\_\_\_2020 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Томск 2020 г.

**Цель работы:** Рассмотреть и составить программы с использованием двумерных массивов и файлов.

**Теоретическая часть**

Двумерные массивы в Паскале трактуется как одномерный массив, тип элементов которого также является массивом (массив массивов). Положение элементов в двумерных массивах Паскаля описывается двумя индексами. Их можно представить в виде прямоугольной таблицы или матрицы.

Описание двумерного массива Паскаля.

Существует несколько способов объявления двумерного массива Паскаля.

Мы уже умеем описывать одномерные массивы, элементы которых могут иметь любой тип, а, следовательно, и сами элементы могут быть массивами. Рассмотрим следующее описание типов и переменных:

Пример описания двумерного массива Паскаля

Program lab5;

Type

Vector = array [1..5] of <тип\_элементов>;

Matrix= array [1..10] of vector;

Var m: matrix;

Мы объявили двумерный массив Паскаля m, состоящий из 10 строк, в каждой из которых 5 столбцов. При этом к каждой i -й строке можно обращаться m [ i ], а каждому j -му элементу внутри i -й строки – m [ i , j ].

Определение типов для двумерных массивов Паскаля можно задавать и в одной строке:

Program lab5;

Type

Matrix= array [1..5] of array [1..10] of < тип элементов >;

или еще проще:

Program lab5;

type

matrix = array [1..5, 1..10] of <тип элементов>;

Обращение к элементам двумерного массива имеет вид: M [ i , j ]. Это означает, что мы хотим получить элемент, расположенный в i -й строке и j -м столбце. Тут главное не перепутать строки со столбцами, а то мы можем снова получить обращение к несуществующему элементу. Например, обращение к элементу M [10, 5] имеет правильную форму записи, но может вызвать ошибку в работе программы.

**Практическая часть**

**Задание 2**

**Исходные данные**: двумерный массив *a(3, 3), состоящий из случайных целых чисел от 1 до 10.*

**Задание**

Вычислить:

* Сумму элементов первой и последней строк массива.

Заполнить массив (3, 3) случайными целыми числами от 1 до 10. Результаты вычислений вывести в файл.

**Программная реализация**

**Program** Lab5;

**var**

i, j: integer;

s: real;

a: **array** [1..3, 1..3] **of** real;

f: text;

**begin**

assign(f, 'result.txt');

rewrite(f);

s := 0;

**for** i := 1 **to** 3 **do**

**begin**

**for** j := 1 **to** 3 **do**

**begin**

a[i, j] := random(10);

write(f, a[i, j]:5);

**end**;

writeln(f)

**end**;

s := s + a[1, i]+ a[3, i];

writeln(f, 'Сумма элементов ', s);

close(f)

**end**.

**Ответ**

4 3 5

6 8 7

4 8 1

Сумма элементов: 25

**Задание 3**

**Исходные данные**: двумерный массив *a (4, 4), состоящий из случайных чисел от -3 до 6.*

**Задание**

Вычислить:

* Среднее арифметическое значений неотрицательных элементов каждого столбца данного массива.

Заполнить массив а (4, 4) случайными числами от -3 до 6. Вывести значения его элементов в файл. Результаты вычислений вывести в файл.

**Программная реализация**

**Program** Lab5;

**var**

i, j, S, n: integer;

a: **array** [1..4, 1..4] **of** integer;

b: **array** [1..4] **of** real;

f: text;

**begin**

assign(f, 'result.txt');

rewrite(f);

**for** i := 1 **to** 4 **do**

**begin**

**for** j := 1 **to** 4 **do**

**begin**

a[i, j] := random(10) - 3;

write(f, a[i, j]:5);

**end**;

Writeln(f);

**end**;

**for** j := 1 **to** 4 **do**

**begin**

**for** i := 1 **to** 4 **do**

**if** a[i, j] >= 0 **then**

**begin**

S := S + a[i, j];

n := n + 1;

**end**;

**if** n > 0 **then** B[j] := S / n **else** B[j] := 0;

Writeln(f, 'Столбец ', j, ' = ', B[j]);

S := 0;

n := 0;

**end**;

close(f);

**end**.

**Ответ**

5 -2 2 3  
2 -1 2 -2  
0 2 3 -3  
4 2 1 1  
Столбец 1 = 2.75  
Столбец 2 = 2  
Столбец 3 = 2  
Столбец 4 = 2

**Задание 4**

**Исходные данные**: двумерный массив *a (4, 4), состоящий из случайных целых чисел от 1 до 100.*

**Задание**

Вычислить:

* Максимальный элемент в каждой строке;
* Минимальный элемент среди максимальных элементов каждой строки.

Заполнить массив а (4, 4) случайными целыми числами от 1 до 100. Вывести значения его элементов в файл. Результаты вычислений вывести в файл.

**Программная реализация**

**Program** Lab5;

**var**

i, j: integer;

max, min: real;

a: **array** [1..4, 1..4] **of** real;

f: text;

**begin**

assign(f, 'result.txt');

rewrite(f);

**for** i := 1 **to** 4 **do**

**begin**

**for** j := 1 **to** 4 **do**

**begin**

a[i, j] := random(100) + 1;

write(f, a[i, j]:5);

**end**;

writeln(f);

**end**;

**for** i := 1 **to** 4 **do**

**begin**

max := a[i, 1];

**for** j := 2 **to** 4 **do**

**if** a[i, j] > max **then**

max := a[i, j];

**if** i = 1 **then** min := max

**else**

**if** max < min **then**

min := max

**end**;

writeln(f);

writeln(f, 'Минимальный среди максимальных ', min);

close(f)

**end**.

**Ответ**

54 37 20 60  
50 8 9 90  
27 82 21 14  
61 14 80 58  
  
Минимальный среди максимальных: 60

**Выводы**

В ходе работы мы рассмотрели и составили программы с использованием двумерных массивов и файлов.