

Урок 6. Работа с табличными данными

Приведенные ранее программы не требовали хранения большого объема данных, поэтому можно было обойтись несколькими переменными. Но бывают задачи, где требуется обрабатывать большое количество однотипной информации. Например, подобные программы работают на компьютерах, сопровождающих соревнования спортсменов.

Обычно эти данные организованы в виде таблиц – матриц (или иначе говорят – двумерных массивов). Простейший вариант таблицы состоит из одной строки или столбца – одномерный массив.

Термин размерность в математике и информатике занят и означает не размер, а количество измерений. Например, массивы с одним индексом называют одномерными (размерности 1), массивы с двумя индексами называют двумерными (размерности 2) и т.д.

Массив – это однотипный набор данных, имеющих общее имя и различающихся порядковым номером. Этот номер называется **индексом элемента массива**. **Имя массива** записывается по таким же правилам, что и имя обычной величины.

Массив может состоять как из целых или действительных чисел, так и принимать символьные значения.

Для того чтобы можно было работать с массивом в программе, надо сделать соответствующее указание в начале программы. Для этого в языке КуМир используется описание таблицы, например **цел таб A[1:10]**

Где A – имя массива, а в квадратных скобках указаны границы индекса массива, то есть наименьшее и наибольшее значения индекса.

Таким образом, описание **цел таб A[1:10]** определяет массив целых чисел, где элементы могут иметь индексы 1, 2, 3 ... 9, 10. То есть этот массив состоит из 10 элементов.

Для описания массива вещественных или символьных данных используются соответствующие описания (**вещ таб A[1:10]**, **лит таб A[1:10]**)

Например, чтобы присвоить пятому элементу массива значение 3, надо в программе написать так:

A[5] := 3.

А оператор **A[7] := A[1] + A[10]** присвоит седьмому элементу массива сумму первого и десятого элементов.

Для вывода значения первого и седьмого элементов массива запишем в алгоритме так:

вывод нс, A[1], A[7]

Индекс можно записывать и как арифметическое выражение, только надо учитывать, что вычисленное значение этого выражения должно быть обязательно целым числом и не выходить за границы индекса массива.

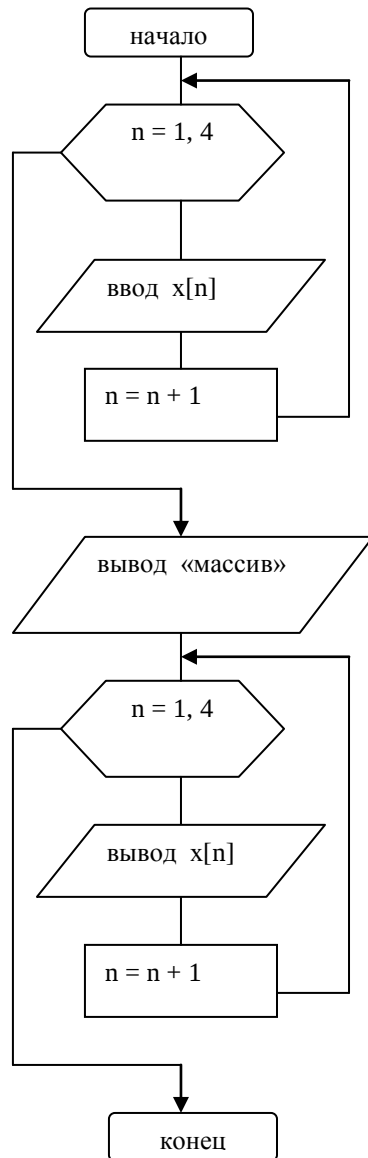
Например, **A[2*K] := A[K+1]**. Если к этому моменту переменная K=4, то в результате действия этого оператора восьмому элементу массива A[8] будет присвоено значение элемента A[5]. А если K=5, то A[10]=A[6].

Итак, значения элементов массива можно задавать с помощью оператора присваивания. Конечно, можно было бы использовать линейный алгоритм, однако если существует какая-либо закономерность в заполнении массива, то гораздо эффективнее применить циклический алгоритм.

Разберем примеры

1. Заполните произвольными числами массив действительных чисел x , размерностью 4, и выведите значение его элементов на экран монитора.

Блок-схема:



Программа

```
алг
нач цел таб x[1:4], цел n
...   нц для n от 1 до 4
...   вывод нс, "введите элемент массива"
...   ввод x[n]
...   кц
вывод нс, "Массив"
нц для n от 1 до 4
...   вывод x[n], " "
...   кц
кон
```

Пояснения

заголовок программы
раздел описания переменных

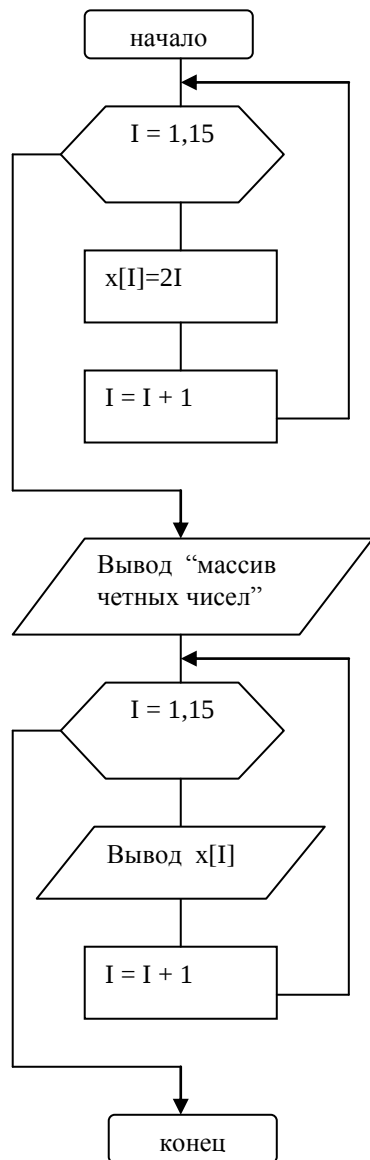
Ввод элемента массива

Вывод заголовка

Вывод массива, элементы
разделяются пробелом

конец алгоритма

2. Заполните массив 15 четными числами и выведите их на экран монитора в одну строку. (Формула четного числа $a=2n$)



алг заполнение массива четными числами

нач **цел** **таб** x[1:15], **цел** I

нц для I от 1 до 15

.....
x[I]:=2*I

кц

вывод **нс**, "МАССИВ ЧЕТНЫХ ЧИСЕЛ"

нц для I от 1 до 15

.....
вывод x[I], " "

кц

кон

Если массив будет достаточно большой, вводить произвольные числа с клавиатуры становится затруднительно. Тем более что это придется делать при каждом запуске программы.

Часто программисты для этой цели используют такую функцию, как **датчик случайных чисел**.

С его помощью можно получать заранее неизвестные, то есть случайные числа и применять их в вычислительных задачах, в игровых программах, при рисовании фигур, размер, цвет и местонахождение которых будет определяться случайным образом.

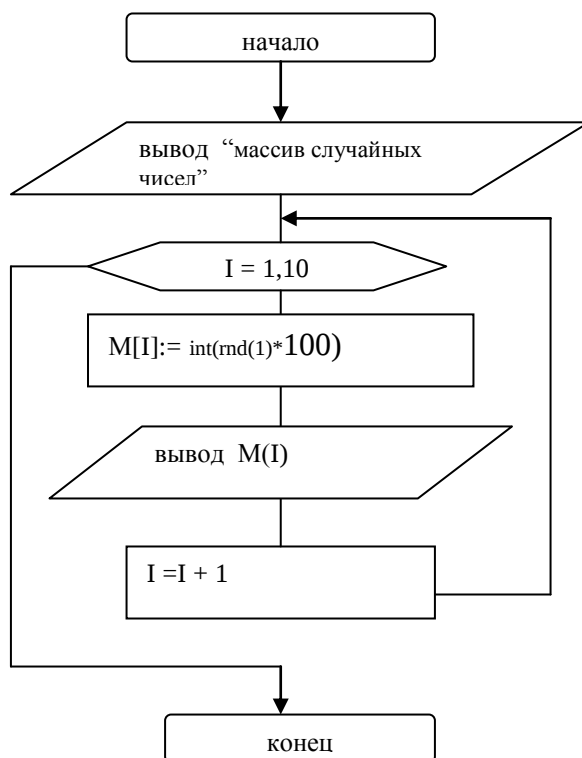
Вызывается эта функция инструкцией **rnd(1)** и выдает случайное значение из промежутка от 0 до 1 (в скобках указан фиктивный параметр 1)

Разберем примеры

1. Заполните массив из 10 элементов целыми случайными числами.

Для заполнения массива целыми числами надо использовать функцию `int`.

Составим блок-схему для этой задачи



алг заполнение массива случайными целыми числами	Название алгоритма
нач цел таб M[1:10], цел i	Описание массива и переменных
вывод "массив случайных чисел"	
нц для I от 1 до 10	Начало цикла
M[I]:= int (rnd(1)*100)	Присваивание элементу массива случайного двузначного числа
вывод M[I], " "	Вывод элемента массива
кц	Конец цикла
кон	Конец алгоритма

2. Массив из 12 элементов заполните случайными целыми числами из промежутка от -5 до 5. Выведите его на экран монитора. Найдите сумму элементов массива и распечатайте на новой строке

Отметим, что случайное число из промежутка [-5; 5] будет задано формулой:

rnd(1)*10 - 5

А для того, чтобы число было обязательно целым, воспользуемся функцией целого числа. Тогда имеем:

int(rnd(1)*10 - 5)

Сумму будем искать в цикле по формуле

S := S + X[I]

Обязательно надо учесть, что в первый момент переменная S должна быть равна 0 .

При первом проходе цикла, будет выполнено: **S := 0 + X[1]**, и в результате сумма после первого шага будет равна первому элементу.

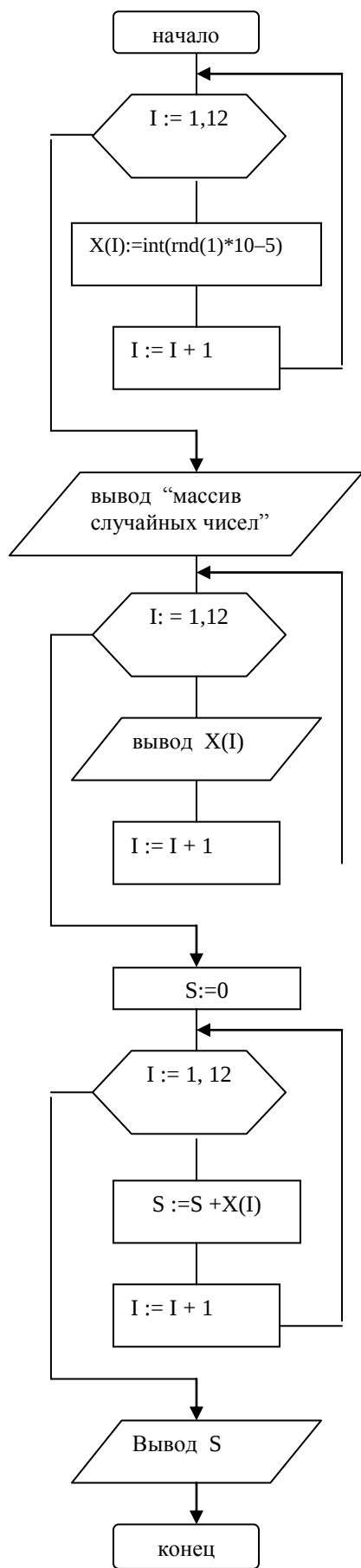
При втором проходе цикла, будет выполнено: **S := S + X[2]**, то есть **X[1] + X[2]**, и в результате сумма после второго шага будет равна сумме первых двух элементов.

При третьем проходе цикла, будет выполнено: **S:= S + X[3]**, иначе говоря **(X[1]+ X[2]) + X[3]**, и в результате сумма после третьего шага будет равна сумме первых трех элементов.

Таким образом, значение переменной S на каждом шаге цикла будет увеличиваться на величину следующего элемента массива.

Теперь напишем программу и блок-схему.

алг сумма элементов массива	заголовок программы
нач цел таб X[1:12], цел I,S	описания переменных
S:=0	Обнуление будущего значения суммы
нц для I от 1 до 12	Начало цикла
X[I]:= int(rnd(1)*10-5)	Присваивание элементу массива случайного числа
S:= S+X[I]	Нахождение суммы
кц	Конец цикла
вывод нс , "МАССИВ СЛУЧАЙНЫХ ЧИСЕЛ"	
нц для I от 1 до 12	Начало цикла
вывод X[I], " "	Вывод элемента массива
кц	Конец цикла
вывод "сумма =", S	Вывод суммы
кон	Конец алгоритма





Выполните задания

Составьте программы и блок-схемы для следующих задач.

1. Заполните массив из 10 элементов значениями последовательности $a=3n^2$. Выведите на первой строке все элементы, на второй строке первый и последний элементы.
2. Заполните массив из 17 элементов случайными целыми числами из промежутка $[-50; 50]$. Выведите на экран сначала все элементы массива, а затем с новой строки элементы с четными индексами.
3. Массив из 5 элементов, заполните произвольными числами с клавиатуры. Выведите на экран элементы с номерами 1, 3, 4.
4. Массив из 15 элементов заполните целыми случайными числами из промежутка $[0; 100]$. Выведите его на экран. Замените значение второго и предпоследнего элементов на 0 и снова выведите массив на экран.
5. Массив из 20 элементов заполните элементами последовательности нечетных чисел. Найдите сумму всех элементов массива.
6. Массив из 25 элементов заполните элементами последовательности $a = \frac{|2n - 10|}{2}$.
Найдите сумму всех элементов массива.
7. Массив из 31 элемента заполните целыми случайными числами из промежутка $[-10; 10]$. Найдите сумму элементов массива, начиная с номера 10 по номер 20.