Решение задач на матрицы.

В языке программирования Pascal матрицы представляют собой двумерные массивы. При обучении программированию решается множество задач на матрицы.

1. ### Сумма элементов двумерного массива

Найти сумму элементов матрицы.

Задача вычисления суммы элементов массива достаточно проста: все элементы массива перебираются и добавляются в одну и ту же переменную. Перебор элементов массива осуществляется в цикле for. Аналогично находится сумма элементов двумерного массива с той лишь разницей, что добавляется вложенный цикл for для прохода по элементам каждой строки матрицы.

2. ### Суммы элементов строк матрицы

Вариант 1. Найти сумму элементов каждой строки матрицы (двумерного массива).

Вариант 2. Вычислить сумму элементов определенной строки матрицы.

Если поставлена задача вычислить сумму элементов каждой строки матрицы, то алгоритм ее выполнения таков:

По-строчно перебираем элементы матрицы (внешний цикл отвечает за переход к новой строке, счетчик - первый индекс элементов).

Во внешнем цикле перед внутренним присваиваем переменной для суммы значение 0. В ней будет накапливаться сумма элементов текущей строки, элементы которой перебираются во внутреннем цикле.

После внутреннего цикла выводим значение переменной-суммы на экран.

Ниже в примере решения данной задачи заполнение матрицы, вывод элементов на экран и подсчет суммы выполняются внутри одного цикла. Это сделано не только для сокращения кода программы, но и

для красивого вывода. После того, как выводятся элементы очередной строки, в конце этой же строки выводится их сумма.

3. Сумма элементов столбцов матрицы

Вариант 1. Вычислить сумму элементов каждого столбца матрицы (двумерного массива).

Вариант 2. Найти сумму элементов определенного столбца матрицы.

Матрицу можно представить как массив, в который вложены другие массивы. Эти другие массивы имеют одинаковую длину (например, M), а количество этих вложенных массивов - другое число (например, N). Так, если M = 7, а N = 5, то это значит, что матрица состоит из 5 одномерных массивов, в каждом из которых по 7 элементов.

Элементы одного вложенного массива обычно считают строкой матрицы. Столбцы же матрицы формируют элементы разных вложенных массивов, но имеющие в них одинаковые индексы (занимающие одинаковые позиции). Так, все первые элементы вложенных массивов, формируют первый столбец матрицы. Элементы с индексом 2 образуют второй столбец.

Если mat - это переменная-матрица, то выражение mat[i,j] обозначает обращение к элементу, имеющему номер строки i (это номер вложенного массива), и номер столбца j (это номер самого элемента во вложенном массиве).

Обычно матрицы заполняются построчно: во внешнем цикле перебираются строки, во внутреннем - элементы строк (формируют столбцы). Однако это не обязательно. Заполнять можно и по столбцам: во внешнем цикле перебирать столбцы, во внутреннем - обращаться к элементам разных вложенных массивов, но имеющих идентичный индекс.

В программе ниже заполнение двумерного массива происходит построчно (стандартно), затем вычисляется сумма элементов каждого столбца, и здесь обход происходит по столбцам. Обратите внимание,

что в данном случае внешний цикл отсчитывает до M, а внутренний - до N. В разных итерациях вложенного цикла различна первая переменная-индекс (в данном случае i), обозначающая номер строки, а столбец остается постоянным.

4. Найти строку матрицы с максимальной суммой элементов

В двумерном массиве (матрице) найти строку, сумма элементов которой является максимальной среди всех строк матрицы.

Решения данной задачи, когда требуется найти только одну строку с максимальной суммой элементов и когда таких строк несколько и все их надо определить, несколько различаются.

Рассмотрим сначала первый случай. Предположим, что в матрице суммы всех строк различны или для решения задачи достаточно найти первую с максимальной суммой.

Пусть переменная sum_max хранит найденную максимальную сумму. Переменная row_max хранит номер строки, элементы которой дают в сумме максимум.

Перебираем построчно матрицу. Перед вложенным циклом переменной sum (хранит сумму элементов текущей строки) присваиваем 0. Во вложенном цикле перебираются элементы текущей строки. Значение каждого из них добавляется к sum.

После того, как сумма элементов текущей строки посчитана (вышли из внутреннего цикла), проверяем не больше ли она значения, хранимого в sum_max. Если больше, то записываем ее в sum_max, а переменной row_max присваиваем номер текущей строки (хранится в переменной i).

В конце программы выводим значения sum_max и row_max (она содержит ответ задачи - номер искомой строки) на экран. В данном случае будет найдена только первая строка с максимальной суммой. Если бы в условном операторе использовался логический оператор >=

(больше или равно), то, при наличии нескольких строк с максимальной суммой, была бы выведена на экран последняя.

5. Найти индексы максимальных элементов матрицы

В двумерном массиве (матрице) найти индексы (номера строк и столбцов) максимальных элементов.

Описание переменных:

mx - заданная матрица;

max - значение максимального элемента;

qty - количество максимальных элементов в матрице;

id - массив для хранения номеров строк и столбцов найденных максимальных элементов;

i, j - переменные, используемые в качестве счетчиков и текущих индексов элементов массива.

Алгоритм решения задачи:

Первым этапом решения данной задачи является поиск значения максимального элемента матрицы. Для этого переменной max сначала присваивается самое минимальное из возможных значений, после этого каждый элемент матрицы сравнивается со значением max. Если текущий элемент больше, то значение max перезаписывается на него. Поиск максимума можно выполнить в том же цикле, в котором происходит заполнения матрицы.

Допустим, задача усложнена тем, что

в матрице может быть несколько равных между собой максимальных элементов,

их индексы нужно не просто вывести на экран, но и сохранить внутри программы.

Если бы требовалось просто вывести индексы первого попавшегося (или единственного) максимального элемента, то достаточно было бы перебрать матрицу, сравнить каждый элемент с ранее найденным максимальным значением. Как только совпадение было бы найдено,

вывести на экран значения i и j. После чего прервать выполнение цикла.

Поскольку нужно запомнить индексы максимумов, то в программу вводится дополнительный двумерный массив (матрица) id. Каждая его строка состоит всего из двух элементов, в которых будут храниться номера строки и столбца найденного максимума. Количество строк массива id должно быть таким, чтобы была возможность записать в него индексы всех элементов заданной матрицы, если вдруг все они окажутся максимальными. Однако маловероятно, что все элементы матрицы будут максимальными. Поэтому массив id не будет заполнен полностью. Чтобы отслеживать сколько строк этого массива заполнено (т. е. сколько максимумов найдено), вводится переменная qty.

Итак, перебираем заданную матрицу построчно. Если очередной элемент равен значению max, то увеличиваем на 1 значение qty, а в id в его строку с номером qty записываем индексы i и j найденного максимума.

В конце программы выводим значения заполненных ранее ячеек массива id.

6. ### Максимальные элементы столбцов матрицы

Найти максимальный элемент для каждого столбца матриц размерностью N строк M столбцов.

При переборе элементов матрицы по столбцам во внешнем цикле изменяется второй индекс, а во внутреннем - первый. Перед внутренним циклом предполагается, что первый элемент столбца двумерного массива - максимальный элемент данного столбца. Во внутреннем цикле, если обнаруживается, что текущий элемент больше уже найденного максимального, то значение переменной перезаписывается на него.

Найденный максимум сразу выводится на экран или может быть сохранен в отдельном одномерном массиве, размерность которого соответствует количеству столбцов матрицы.

7. Поменять местами строки матрицы
8. Отсортировать в матрице столбцы по убыванию значений элементов в первой строке
9. Найти максимальный элемент диагонали
10. Количество отрицательных элементов под главной диагональю матрицы
11. Найти минимальный элемент матрицы ниже побочной диагонали
12. Заполнить двумерный массив по правилу
13. Умножение матриц

Источник https://pas1.ru/taskmatrix