УДК 004.432.2

ДИНАМИЧЕСКИЕ МАССИВЫ НА С++ И ЗАДАЧА ПОИСКА МИНИМАКСОВ

Петрова Наталья Константиновна канд. физ.-мат. наук, доцент кафедры информатики и информационно-управляющих систем, ФГБОУ ВО «Казанский государственный энергетический университет» (Россия, Казань)
Алабдуллах Мохамад Белал студент группы ПОВТ 1-16, ФГБОУ ВО «Казанский государственный энергетический университет» (Россия, Казань)
Альхузайи Ахмед Хушам Хусам студент группы ПОВТ 1-16, ФГБОУ ВО «Казанский государственный энергетический университет» (Россия, Казань) **Каттави Раджих Зиад**
студент группы ПОВТ 1-16, ФГБОУ ВО «Казанский государственный энергетический университет» (Россия, Казань)

В представленной статье рассматривается классическая задача поиска седловой точки в двумерном массиве. Подобного рода задачи весьма востребованы в геодезии, при построении топографических или гравиметрических карт местности, в энергетике – при планировании нагрузки на электрические или тепловые сети. Код программ предлагаемого проекта написан в рамках структурного подхода на языке С++ в среде разработки VisualStudio 2017.

Ключевые слова: седловая точка, VSStudio, C++, динамические массивы.

DYNAMIC ARRAYS WITH C ++ AND THE PROBLEM OF SEARCH FOR MINIMAX

Petrova Natalya Konstantinovna
cand. phys.-mat. sciences, associate professor of the department of informatics and information management systems, Kazan State Energy University (Russia, Kazan)
Alabdullah Mohamad Belal
student of the POVT 1-16 group, FSBEI of HE "Kazan State Energy University" (Russia, Kazan)
Alhuzai Ahmed Husham Hussam
student of the POVT 1-16 group, Kazan State Energy University (Russia, Kazan)
Kattavi Radzhih Ziad
student of the POVT 1-16 group, Kazan State Energy University (Russia, Kazan)

The article presents the classical problem of finding a saddle point in a two-dimensional array. Such tasks are highly demanded in geodesy, in the construction of topographic or gravimetric maps of the terrain, in the energy sector - in planning the load on electric or thermal networks. The program code of the proposed project is written within the framework of the structured approach in the C ++ language in the Visual Studio 2017 development environment.

Keywords: saddle point, VS Studio, C ++, dynamic arrays.

Введение. Возможности операций с динамическими массивами на C++ позволяет без особых затруднений решать классические задачи на их основе для закрепления навыков

выполнения низкоуровневых операций при работе с отдельными элементами массива, а также передаче массивов в другие функции через ссылки и указатели, строя проект в рамках технологии структурного программирования. Дадим определение ключевомупонятию «седловая точка», рассматриваемого в нашей статье. Седловая точка - элемент матрицы, являющуюся наименьшим в своей строке и одновременно наибольшим в своем столбце или, наоборот, является наибольшим в своей строке и наименьшим в своем столбце. Составим программный проект, который для любого двумерного массива размера $(N \times M)$, где *N*–количество строк, *M*–количество столбцов,находит индексы седловых точек и их количество.Программу будем реализовывать в среде программирования VisualStudio 2017 на языке C++.

Контрольные примеры для разработки алгоритма и обеспечения полноты программы.Для реализации проекта необходимо начать разработку с составления контрольных примеров.В качестве контрольного примера для тестирования рассмотрим двумерный массив размером (n = 3; m = 4), заполним его числами в пределах [0; 100] (рис. 1).

	0	1	2	3
0	45	84	25	17
1	60	72	33	1
2	80	97	47	29

Рис. 1. Исходный массив

Далее найдем максимальные элементы в каждой строке (рис. 2):

	0	1	2	3
0	45	84	25	17
1	60	72	33	1
2	80	97	47	29

Рис. 2. Максимальные элементы строк

Далее найдем минимальные элементы в каждой строке (рис. 3):

	0	1	2	3
0	45	84	25	17
1	60	72	33	1
2	80	97	47	29

Рис. 3. Минимальные элементы строк

Далее найдем максимальные элементы в каждом столбце (рис. 4):

	0	1	2	3
0	45	84	25	17
1	60	72	33	1
2	80	97	47	29

Рис. 4. Максимальные элементы столбцов

Далее найдем минимальные элементы в каждом столбце (рис. 5):

	0	1	2	3
0	45	84	25	17
1	60	72	33	1
2	80	97	47	29

Рис. 5. Минимальные элементы столбцов

Далее сравниваем максимальные элементы строк и минимальные элементы столбцов, и наоборот, минимальные

элементы строк и максимальные элементы столбцов, если они совпадают, то записываем индексы этой точки и увеличиваем количество седловых точек на 1.

Проанализировав результаты, видим, что есть 2 *седловые точки* с индексами (1; 1) и (2; 3)(рис. 6):

	0	1	2	3
0	45	84	25	17
1	60	72	33	1
2	80	97	47	29

Рис. 6. Седловые точки

Анализ контрольных примеров показывает, что решение данной задачи требует использование следующих алгоритмических методов:

1. Введение счѐтчика. Переменная k для подсчета количества седловых точек.
2. Использование операторов ввода/вывода. Операторы *cin* и *cout*.
3. Использование циклических алгоритмов: *for* и *while*.
4. Работа с индексами двумерного массива. *N* – количество строк, *M*–количество столбцов массива *a*.
5. Введение временных переменных. *Min* – минимальный элемент строки; *Max* – максимальный элемент строки; *MinCol* – минимальный элемент столбца; *MaxCol* – максимальный элемент столбца.
6. Нахождение максимума.
7. Нахождение минимума.

8. Передача массива в функции. Массив передаѐтся в функции через указатели на указатели. Функция *MakeArr* – создает массив, *print*–выводит массив, *solution*–вычисляет седловые точки массива и выводит их индексы на экран.

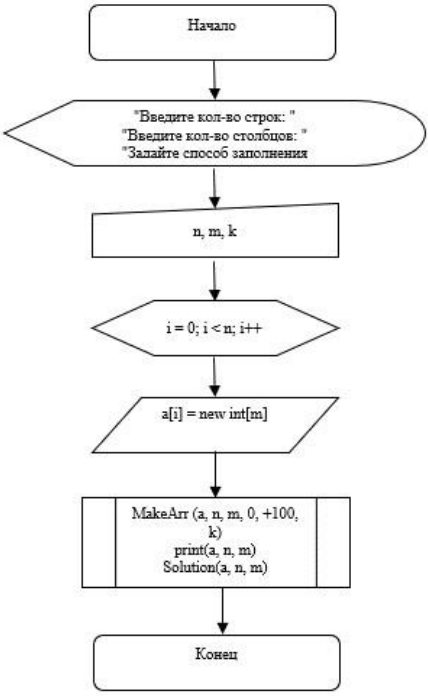


Рис. 7. Главная программа

В качестве примера приведем текст функции MakeArr:

```
void MakeArr(int **tf, int nf, int mf, int r_min, int r_max, int way)
{
        srand((unsigned int)time(NULL));
```

```
        for (int i = 0; i < nf; i++)
for (int j = 0; j < mf; j++)
                {
                        if (way == 1)
                                tf[i][j] = rand() % (r_max - r_min) + r_min;
                        else
                        {                                               cout <<
"Элемент (" << i << "; " << j << "):
";
                                cin >> tf[i][j];
                        }
                }

        return;
}
```

Inttf – исходный массив; *intnf* – количество строк; *intmf* – количество столбцов; *intr_min* – левая граница генерируемых чисел; *intr_max* – правая граница генерируемых чисел; *intway* – флаг, введенный для выбора способа инициализации массива. Если он равен 1, то инициализация идѐт с помощью генератора случайных чисел в заданном диапазоне значений от *r_min* до *r_max*, в противном случае инициализация осуществляется вводом значений с консоли.

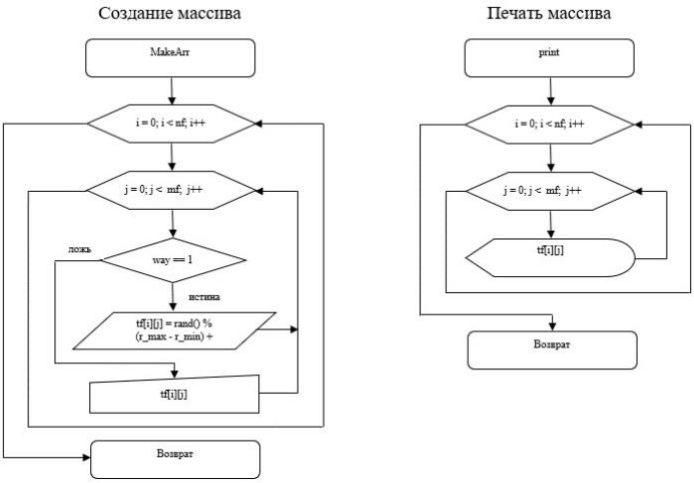


Рис. 8. Создание и печать массива

Заключение. По итогам проведенной работы нами разработан алгоритм, использующий приемы структурного программирования, для нахождения седловых точек в двумерном массиве, визуализированный словесно-блочной схемой, разработан контрольный пример для проверки корректности программного кода.Программу протестировали 3 раза:

1) На 2 способа инициализации двумерного

массива. (1-2)

2) На неправильно введенный размер двумерного массива. (3)

Результаты отладки программыв среде разработки VisualStudio 2017представлены ниже:

1) Способ инициализации вручную с консоли

$k = 0$.

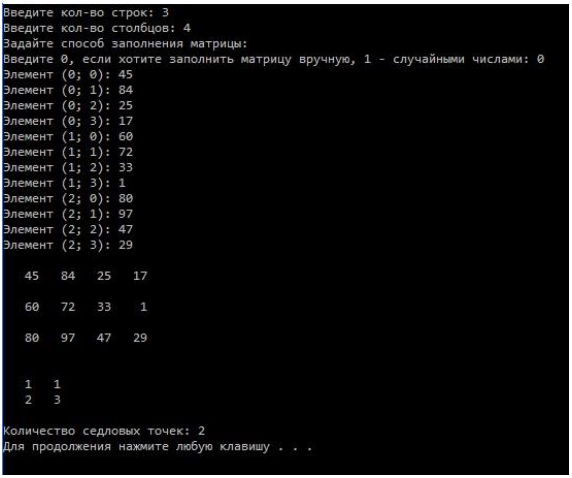


Рис. 9. Способ инициализации вручную с консоли

2) Способ инициализации случайными числами $k = 1$.

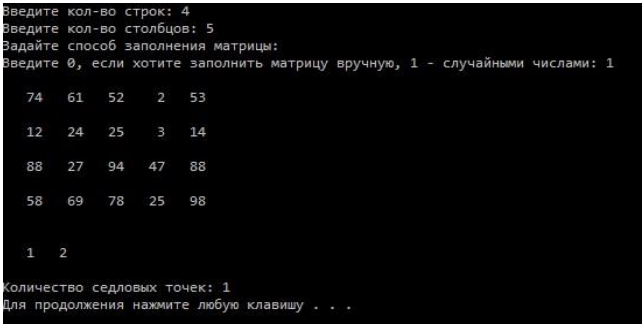


Рис. 10. Способ инициализации случайными числами

3) Неправильно введенный размер двумерного массива: $n = 0, m < 0$.

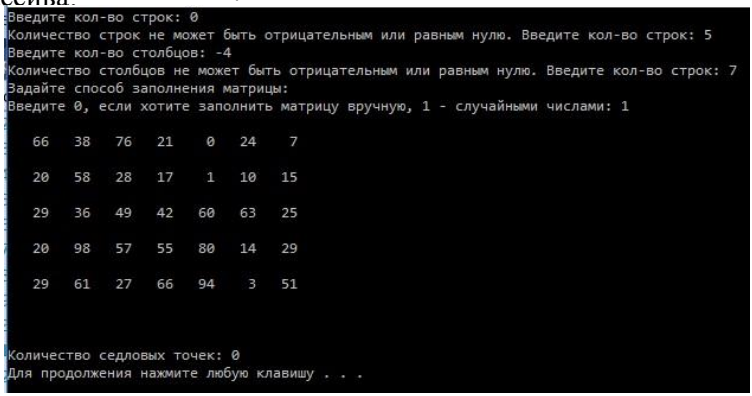


Рис. 11. Проверка на неправильно введенные данные

Все седловые точки совпадают с представленным контрольным примером.

Результат: 1) -2 седловые точки с индексами (1; 1) и (2; 3)(рис. 9). Результат: 2) - 1 седловая точка с индексом (1; 2) (рис. 10).

Результат: 3) - проверка на неправильно введенные данные, седловые точки отсутствуют (рис. 11).

Список литературы

1. Петрова Н.К. Конспект лекций «Программирование на С++». Электронный документ https://lms.kgeu.ru/course/view.php?id=2594.

2. Довбуш, Галина Visual C++ на примерах / Галина Довбуш , Анатолий Хомоненко. - М.: БХВ-Петербург, 2012. - 528 с.

3. Балена, Франческо Современная практика программирования на MicrosoftVisualBasic и Visual C# Русская Редакция, 2015. - 640 с.

4. Лафоре, Роберт Объектно-ориентированное программирование в C++ 4-е издание Питер, 2018. - 928 с.

5. Сидорина, Татьяна Самоучитель MicrosoftVisualStudio C++ и MFC / Татьяна Сидорина. - М.: "БХВ-Петербург", 2014. - 848 с.

© Петрова Н.К., Алимов Т.М., Алабдуллах М.Б., Альхузайи А.Х.Х, Каттави Р.З., 2019