

**Ткаченко Кирилл Станиславович***Инженер 1-й кат.**ФГАОУ ВО «Севастопольский
государственный университет».*

Разработка программы для построения магических квадратов нечетного порядка на школьном алгоритмическом языке

Рассматривается разработка программы для построения магических квадратов нечетного порядка на школьном алгоритмическом языке. Приводятся полный исходный текст и результаты работы программы. Программа демонстрирует использование основных конструкций школьного алгоритмического языка, работу с подпрограммами и массивами, поэтому может быть полезна всем, изучающим информатику и ИКТ, программирование и школьный алгоритмический язык.

Издревле известны так называемые магические квадраты [1, 2]. Магическим квадратом называется квадратная матрица (таблица), каждый элемент которой является целым положительным (натуральным) числом, уникальным в рамках магического квадрата, причем суммы элементов каждой строки, столбца, главной и побочной диагонали равны между собой. Количество элементов магического квадрата по строке или столбцу называется его порядком. Обычно магические квадраты заполняются значениями $1, 2, \dots, \text{Порядок}^2$.

Например, магический квадрат 3-го порядка:

8	1	6
3	5	7
4	9	2

Главная диагональ – это элементы 8, 5, 2. Побочная диагональ – это элементы 6, 5, 4. Суммы элементов по строкам, столбца и диагоналям равны магической константе 15:

8	1	6	15
3	5	7	15
4	9	2	15
15	15	15	

Для магических квадратов произвольного порядка известна формула для расчета магической константы [1, 2]:

$$\text{МагическаяКонстанта} = \frac{\text{Порядок}(\text{Порядок}^2 + 1)}{2}. \quad (1)$$

Способы построения магических квадратов могут различаться для различных значений порядка. В настоящей работе далее рассматривается подход, пригодный исключительно для нечетных порядков.

Поскольку магические квадраты широко известны, то их программное построение является очень хорошей задачей, решенной на различных языках программирования [3]. Целью настоящей работы является разработка программы для построения магических квадратов нечетного порядка на школьном алгоритмическом языке.

Программа начинается с алгоритма Потенциал, который организует построение (попытку построения) магических квадратов нечетного порядка для порядков 2, 4, 3, 5, 7 (четные порядки приведены для проверки существующих ограничений):

```
алг Потенциал
нач
  Построить (2) ;
  Построить (4) ;
  Построить (3) ;
  Построить (5) ;
  Построить (7) ;
кон
```

Далее следует алгоритм Построить(цел Порядок), который для заданного порядка Порядок выполняет попытку построения магического квадрата нечетного порядка. В целочисленном двумерном массиве Квадрат располагается результирующий магический квадрат. В целочисленных переменных находятся, соответственно, МагическаяКонстанта – магическая константа, Сумма – проверочные суммы, А, Б, В – счетчики циклов и индексы:

```
алг Построить (цел Порядок)
нач
  цел таб Квадрат [1 : Порядок, 1 : Порядок] ;
  цел МагическаяКонстанта ;
  цел Сумма ;
  цел А, Б, В ;
```

Если порядок не является нечетным числом, не меньшим 3, то демонстрируемый алгоритм не может работать:

```
если Порядок < 3 или mod (Порядок, 2) = 0
  то
    вывод "Неверный порядок ", Порядок, ". ", нс, нс ;
    выход ;
все
```

По формуле (1) рассчитывается магическая константа, затем она отображается:

```
МагическаяКонстанта := div(Порядок * (Порядок * Порядок + 1), 2);  
вывод "Магическая константа для ", Порядок, " на ", Порядок,  
" есть ", МагическаяКонстанта, нс;
```

Магический квадрат инициализируется нулями:

```
нц для А от 1 до Порядок  
  нц для Б от 1 до Порядок  
    Квадрат[А, Б] := 0;  
  кц  
кц
```

Теперь непосредственно выполняется построение магического квадрата. Начиная с середины первой строки происходит заполнение массива элементами со значениями от 1 до Порядок^2 с шагом 1:

```
А := 1;  
Б := div(Порядок, 2) + 1;  
нц для В от 1 до Порядок * Порядок  
  Квадрат[А, В] := В;
```

В случае, если имеется необходимость произвести смену только для строки, то новый номер рабочей строки рассчитывается:

```
если Квадрат[mod(А + Порядок - 2, Порядок) + 1, mod(Б +  
Порядок, Порядок) + 1] <> 0  
  то  
    А := mod(А + Порядок, Порядок) + 1;
```

Иначе рассчитываются новые номера рабочих строки и столбца:

```
иначе  
  А := mod(А + Порядок - 2, Порядок) + 1;  
  Б := mod(Б + Порядок, Порядок) + 1;  
все  
кц
```

Магический квадрат отображается:

```
нц для А от 1 до Порядок  
  нц для Б от 1 до Порядок  
    вывод Квадрат[А, Б] : 3;  
  кц  
  вывод нс;  
кц  
вывод нс;
```

Для каждой строки полученного магического квадрата производится проверка корректности его построения путем суммирования элементов, принадлежащих рассматриваемой строке, сумма сообщается пользователю:

```
нц для А от 1 до Порядок  
  Сумма := 0;  
  нц для Б от 1 до Порядок  
    вывод Квадрат[А, Б] : 3;  
    Сумма := Сумма + Квадрат[А, Б];  
  кц
```

```
    вывод " : ", Сумма, " строка ", А, нс;  
кц
```

Для каждого столбца полученного магического квадрата производится проверка корректности его построения путем суммирования элементов, принадлежащих рассматриваемому столбцу, сумма сообщается пользователю:

```
нц для В от 1 до Порядок  
    Сумма := 0;  
    нц для А от 1 до Порядок  
        вывод Квадрат[А, В] : 3;  
        Сумма := Сумма + Квадрат[А, В];  
    кц  
    вывод " : ", Сумма, " столбец ", В, нс;  
кц
```

Для главной диагонали полученного магического квадрата производится проверка корректности его построения путем суммирования элементов, принадлежащих главной диагонали, сумма сообщается пользователю:

```
Сумма := 0;  
А := 1;  
В := 1;  
нц пока А <= Порядок  
    вывод Квадрат[А, В] : 3;  
    Сумма := Сумма + Квадрат[А, В];  
    А := А + 1;  
    В := В + 1;  
кц  
вывод " : главная диагональ ", Сумма, нс;
```

Для побочной диагонали полученного магического квадрата производится проверка корректности его построения путем суммирования элементов, принадлежащих побочной диагонали, сумма сообщается пользователю:

```
Сумма := 0;  
А := Порядок;  
В := 1;  
нц пока В <= Порядок  
    вывод Квадрат[А, В] : 3;  
    Сумма := Сумма + Квадрат[А, В];  
    А := А - 1;  
    В := В + 1;  
кц  
вывод " : побочная диагональ ", Сумма, нс;
```

Алгоритм завершается:

```
    вывод нс;  
кон
```

Схема алгоритма Построить приводится на рисунке 1 и 2, при этом на рисунке 1 приводится фрагмент алгоритма до отображения магического квадрата включительно, а на рисунке 2 – начиная с проверки корректности построения магического квадрата по строкам.

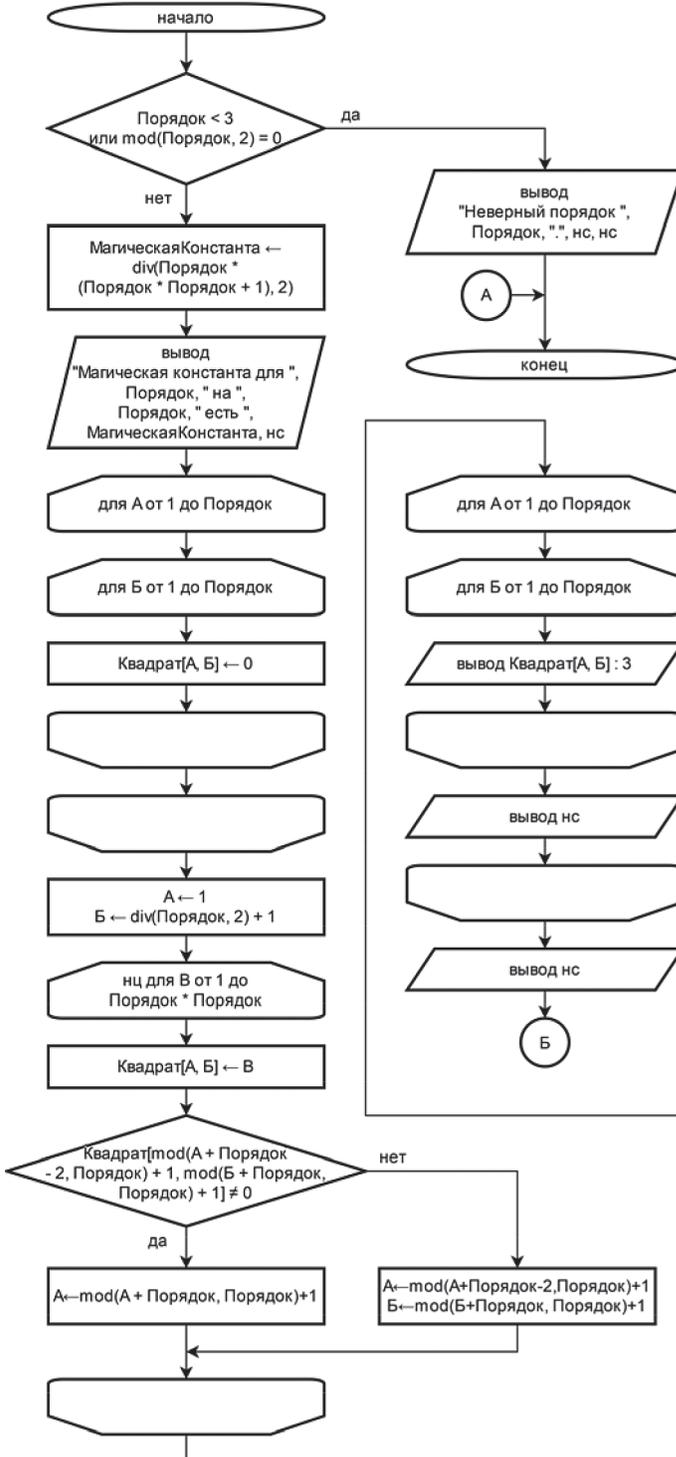


Рис.1

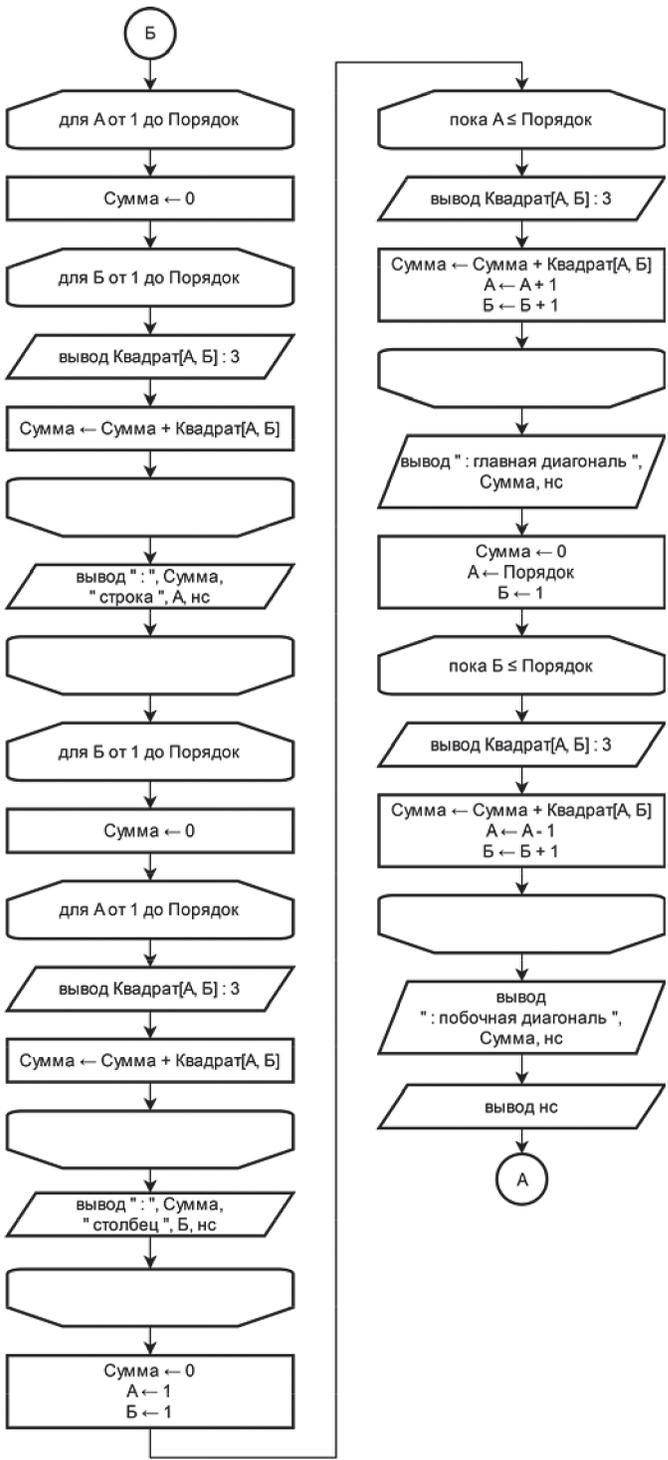


Рис. 2

Полный исходный текст программы находится в приложении А, результаты работы – приложении Б, программа в системе КуМир – на снимке экрана.

Разработанная программа является хорошей демонстрацией использования основных конструкций школьного алгоритмического языка, в том числе, работы с подпрограммами и массивами, поэтому может быть полезна всем, кто изучает информатику и ИКТ, программирование и школьный алгоритмический язык.

Список литературы

1. Магический квадрат // Википедия. URL: https://ru.wikipedia.org/wiki/Магический_квадрат (дата обращения: 03.12.2020).
2. Magic square // Wikipedia. URL: https://en.wikipedia.org/wiki/Magic_square (дата обращения: 03.12.2020).
3. Magic squares of odd order // Rosetta Code. URL: http://rosettacode.org/wiki/Magic_squares_of_odd_order (дата обращения: 03.12.2020).

Приложение А

Полный исходный текст программы

```
алг Потенциал
```

```
нач
```

```
    Построить (2) ;
```

```
    Построить (4) ;
```

```
    Построить (3) ;
```

```
    Построить (5) ;
```

```
    Построить (7) ;
```

```
кон
```

```
алг Построить (цел Порядок)
```

```
нач
```

```
    цел таб Квадрат [1 : Порядок, 1 : Порядок] ;
```

```
    цел МагическаяКонстанта ;
```

```
    цел Сумма ;
```

```
    цел А, В, В ;
```

```
    если Порядок < 3 или mod(Порядок, 2) = 0
```

```
        то
```

```
            вывод "Неверный порядок ", Порядок, ".", нс, нс ;
```

```
            выход ;
```

```
        все
```

```
        МагическаяКонстанта := div(Порядок * (Порядок * Порядок + 1), 2) ;
```

```
        вывод "Магическая константа для ", Порядок, " на ", Порядок, " есть ", МагическаяКонстанта, нс ;
```

```
    нц для А от 1 до Порядок
```

```
        нц для В от 1 до Порядок
```

```
            Квадрат [А, В] := 0 ;
```

```
        кц
```

```
    кц
```

```
A := 1;
B := div(Порядок, 2) + 1;
нц для B от 1 до Порядок * Порядок
    Квадрат[A, B] := B;

    если Квадрат[mod(A + Порядок - 2, Порядок) + 1, mod(B +
Порядок, Порядок) + 1] <> 0
        то
            A := mod(A + Порядок, Порядок) + 1;

            иначе
                A := mod(A + Порядок - 2, Порядок) + 1;
                B := mod(B + Порядок, Порядок) + 1;
            все
        кц

нц для A от 1 до Порядок
    нц для B от 1 до Порядок
        вывод Квадрат[A, B] : 3;
    кц
    вывод нс;
кц

вывод нс;

нц для A от 1 до Порядок
    Сумма := 0;
    нц для B от 1 до Порядок
        вывод Квадрат[A, B] : 3;
        Сумма := Сумма + Квадрат[A, B];
    кц
    вывод " : ", Сумма, " строка ", A, нс;
кц

нц для B от 1 до Порядок
    Сумма := 0;
    нц для A от 1 до Порядок
        вывод Квадрат[A, B] : 3;
        Сумма := Сумма + Квадрат[A, B];
    кц
    вывод " : ", Сумма, " столбец ", B, нс;
кц

Сумма := 0;
A := 1;
B := 1;
нц пока A <= Порядок
```

```
    вывод Квадрат[A, B] : 3;  
    Сумма := Сумма + Квадрат[A, B];  
    A := A + 1;  
    B := B + 1;  
кц  
вывод " : главная диагональ ", Сумма, нс;  
  
Сумма := 0;  
A := Порядок;  
B := 1;  
нц пока B <= Порядок  
    вывод Квадрат[A, B] : 3;  
    Сумма := Сумма + Квадрат[A, B];  
    A := A - 1;  
    B := B + 1;  
кц  
вывод " : побочная диагональ ", Сумма, нс;  
  
    вывод нс;  
кон
```

Приложение Б

Результаты работы программы

Неверный порядок 2.

Неверный порядок 4.

Магическая константа для 3 на 3 есть 15

```
8  1  6  
3  5  7  
4  9  2
```

```
8  1  6 : 15 строка 1  
3  5  7 : 15 строка 2  
4  9  2 : 15 строка 3  
8  3  4 : 15 столбец 1  
1  5  9 : 15 столбец 2  
6  7  2 : 15 столбец 3  
8  5  2 : главная диагональ 15  
4  5  6 : побочная диагональ 15
```

Магическая константа для 5 на 5 есть 65

```
17 24  1  8 15  
23  5  7 14 16  
 4  6 13 20 22  
10 12 19 21  3  
11 18 25  2  9
```

17 24 1 8 15 : 65 строка 1
23 5 7 14 16 : 65 строка 2
4 6 13 20 22 : 65 строка 3
10 12 19 21 3 : 65 строка 4
11 18 25 2 9 : 65 строка 5
17 23 4 10 11 : 65 столбец 1
24 5 6 12 18 : 65 столбец 2
1 7 13 19 25 : 65 столбец 3
8 14 20 21 2 : 65 столбец 4
15 16 22 3 9 : 65 столбец 5
17 5 13 21 9 : главная диагональ 65
11 12 13 14 15 : побочная диагональ 65

Магическая константа для 7 на 7 есть 175

30 39 48 1 10 19 28
38 47 7 9 18 27 29
46 6 8 17 26 35 37
5 14 16 25 34 36 45
13 15 24 33 42 44 4
21 23 32 41 43 3 12
22 31 40 49 2 11 20

30 39 48 1 10 19 28 : 175 строка 1
38 47 7 9 18 27 29 : 175 строка 2
46 6 8 17 26 35 37 : 175 строка 3
5 14 16 25 34 36 45 : 175 строка 4
13 15 24 33 42 44 4 : 175 строка 5
21 23 32 41 43 3 12 : 175 строка 6
22 31 40 49 2 11 20 : 175 строка 7
30 38 46 5 13 21 22 : 175 столбец 1
39 47 6 14 15 23 31 : 175 столбец 2
48 7 8 16 24 32 40 : 175 столбец 3
1 9 17 25 33 41 49 : 175 столбец 4
10 18 26 34 42 43 2 : 175 столбец 5
19 27 35 36 44 3 11 : 175 столбец 6
28 29 37 45 4 12 20 : 175 столбец 7
30 47 8 25 42 3 20 : главная диагональ 175
22 23 24 25 26 27 28 : побочная диагональ 175

Юмор Юмор Юмор Юмор Юмор Юмор

* * *

Компьютерная программа делает то, что вы ей приказали делать, а не то, что бы вы хотели, чтобы она делала.