



Ткаченко Кирилл Станиславович

*Специалист по учебно-методической работе
аспирант, ассистент*

Севастопольского государственного университета.

Пример решения интересной классической рекурсивной задачи итерационным методом на школьном алгоритмическом языке

Рассматривается реализованная на школьном алгоритмическом языке программа для итерационного нахождения всех решений задачи о восьми ферзях. Реализация позволит разобрать работу с операторами цикла и выбора, одномерными массивами. Приводятся полный текст программы и результаты её работы.

Современные программные системы и комплексы становятся всё более сложными и ресурсоёмкими. Поэтому изучающим программирование необходимо приобретать умения по разработке эффективных программ. Хорошим примером такой программы является программа, предназначенная для итеративного отыскания всех решений задачи о восьми ферзях [1], хотя классическим считается рекурсивный подход [2]. Это решение может быть реализова-

но программно как на высокоуровневых языках [3], так и на популярном языке автоматизации 1С [4]. Целью настоящей работы является разработка такой программы на школьном алгоритмическом языке.

Собственно, сама задача о восьми ферзях [1, 2] на доске 8×8 сводится к нахождению таких размещений восьми ферзей на этой доске, при которых ни один из ферзей не бьёт никакого другого ферзя по горизонтали, вертикали и диагонали.

Такая постановка задачи приводит к тому, что на одной вертикали способен находиться исключительно один ферзь. Поэтому для представления шахматной доски в данном случае достаточно одномерного массива, первый элемент которого содержит горизонталь ферзя с вертикалью «а», второй элемент содер-

жит горизонталь ферзя с вертикалью «b», ..., восьмой элемент содержит горизонталь ферзя с вертикалью «h». Непосредственно номера горизонталей, содержащиеся в этом массиве, нумеруются от 1 до 8.

Основной алгоритм выполняет только вызов алгоритма, осуществляющего поиск решений:

```
алг Потенциал
нач
  РешениеЗадачиОВосьмиФерзях;
кон
```

Алгоритм для поиска решений задачи о восьми ферзях не имеет аргументов и не возвращает результата:

```
алг РешениеЗадачиОВосьмиФерзях
нач
```

В начале алгоритма присутствуют объявления переменных. В одномерном массиве **доска** содержится 8 целочисленных элементов для хранения горизонталей ферзей. Целочисленные переменные **i** и **j** хранят

индексы элементов-вертикалей в этом массиве. Булевы переменные **подАтакой** и **возврат** отмечают соответственно наличие как минимум одного ферзя под атакой и завершение поиска решений:

```
цел таб доска[1:8];
цел i, j;
лог подАтакой, возврат;
```

Вначале отмечается незавершенность поиска и номер текущей рас-

сматриваемой вертикали устанавливается в единицу:

```
возврат := нет;
i := 1;
```

В цикле, пока поиск решений не окончен, ферзь устанавливается на

наименьшую допустимую позицию *i*-й вертикали:

```
нц пока не возврат
  доска[i] := 1;
```

Вновь, пока поиск решений не окончен, отмечается отсутствие взаимной атаки ферзей:

```
нц пока не возврат
  подАтакой := нет;
```

В случае рассмотрения вертикали с номером, не равным 1, ферзей больше одного:

```
если i <> 1
то
```

И тогда надо проверить, чтобы все предыдущие ферзи, находящиеся на

вертикалях 1, 2, ..., $i-1$, не могли атаковать ферзя на вертикали i :

```
нц для j от 1 до i - 1
```

Ферзи на разных вертикалях i и j атакуют друг друга, если равны их горизонтали либо равны абсолютные

величины разности горизонталей и абсолютные величины разности вертикалей:

```
если (доска[i] = доска[j]) или
    (abs(доска[i] - доска[j]) = abs(i - j))
то
```

При этом отмечаем нахождение под атакой и покидаем вложенный цикл:

```
подАтакой := да;
выход;
все
кц
```

Если мы находимся на первой вертикали, но номер горизонтали

соответствует верхней половине доски,

```
иначе
если доска[i] >= 5
то
```

то для исключения симметричных решений надо завершать поиск решений:

```
возврат := да;
выход;
все
все
```

Если нет ферзей под атакой, то переходим на следующую вертикаль:

```
если не подАтакой
то
i := i + 1;
```

Если не вышли за пределы доски, то продолжаем поиск решений:

```
если i <= 8
то
выход;
все
```

Состояние доски выводится:

```
ПечатьДоски(доска);
```

Переходим на последнюю вертикаль:

```
    i := 8;  
все
```

Уменьшаем номер вертикали, пока номер горизонтали ферзя на ней

не станет меньше номера последней горизонтали:

```
нц пока доска[i] >= 8  
    i := i - 1;  
кц
```

Сдвигаем ферзя на текущей вертикали на одну горизонталь выше:

```
    доска[i] := доска[i] + 1;  
кц  
кц  
кон
```

Алгоритм печати доски выводит одномерный целочисленный массив на устройство вывода, единственная

целочисленная переменная i играет роль счётчика цикла:

```
алг ПечатьДоски(цел таб доска[1:8])  
нач  
    цел i;
```

Для всех вертикалей выводится номер горизонтали ферзя на каждой из них:

```
нц для i от 1 до 8  
    вывод " ", доска[i];  
кц
```

Строка вывода завершается:

```
    вывод нс;  
кон
```

Полный исходный текст программы на школьном алгоритмическом языке приводится в приложении А, результаты работы программы – в приложении Б.

Полученная программа позволит изучающим программирование разобрать работу с операторами цикла и выбора, одномерными массивами.

Приложение А. Исходный текст программы на школьном алгоритмическом языке

```
алг Потенциал
нач
  РешениеЗадачиОВосьмиФерзях;
кон

алг РешениеЗадачиОВосьмиФерзях
нач
  цел таб доска[1:8];
  цел i, j;
  лог подАтакой, возврат;

  возврат := нет;
  i := 1;

  нц пока не возврат
    доска[i] := 1;

    нц пока не возврат
      подАтакой := нет;

      если i <> 1
        то
          нц для j от 1 до i - 1
            если (доска[i] = доска[j]) или (abs(доска[i] -
доска[j]) = abs(i - j))
              то
                подАтакой := да;
                выход;
          все
        кц

      иначе
        если доска[i] >= 5
          то
            возврат := да;
            выход;
          все
        все

  если не подАтакой
    то
      i := i + 1;
      если i <= 8
        то
          выход;
```

```
все
    ПечатьДоски(доска);
    i := 8;
все

нц пока доска[i] >= 8
    i := i - 1;
кц
доска[i] := доска[i] + 1;
кц
кц
кон

алг ПечатьДоски(цел таб доска[1:8])
нач
    цел i;
    нц для i от 1 до 8
        вывод " ", доска[i];
    кц
    вывод нс;
кон
```

Приложение Б. Результаты работы программы

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

Список литературы

1. *Ламуатье Ж.-П.* Упражнения по программированию на Фортране-IV. – М.: Мир, 1978. – 168 с.
2. *Вирт Н.* Алгоритмы + структуры данных = программы. – М.: Мир, 1985. – 406 с.
3. *Ткаченко К.С.* Интересное решение задачи о восьми ферзях // Потенциал. Математика. Физика. Информатика. 2016. № 7. С. 47–52.
4. *Ткаченко К.С.* Пример решения классической рекурсивной задачи итерационным способом средствами 1С // Системный администратор. 2017. № 7–8 (176–177). С. 84–85.

Ответы к задачам со с. 35

1. 12,5 м/с. 2. 3 м. 3. ≈ 8 с. 4. $\approx 8,4$ с. 5. Сначала включается двигатель с большим ускорением a_1 , а затем – с меньшим a_2 .

Мудрые мысли

Мудрые мысли

Мудрые мысли

Бытиё человека покоится на двух китах: чувствах и знаниях. Чувства без знаний неэффektivны, знания без чувств бесчеловечны.

В. Вайскопф

Самое прекрасное и глубокое из доступных нам чувств – это ощущение тайны, ибо в ней источник науки.

А. Эйнштейн

Естествознание так человечно, так правдиво, что я желаю удачи каждому, кто отдаётся ему.

И.В. Гёте

Побуждал и будет побуждать людей посвящать своё время и труд разработке научных вопросов только прирождённый талант, талант понимать, чувствовать и угадывать стройные соотношения в предвечных законах природы.

П.Н. Лебедев

Талант развивается из чувства любви к делу, возможно даже, что талант – в сущности его – и есть только любовь к делу, к процессу работы.

М. Горький

Облагораживают человека не знания ... а стремление к истине, пробуждающееся тогда, когда человек начинает приобретать знания.

Н.И. Пирогов