**Глава 16. Решение задачи «О восьми ферзях» рекурсивным методом**

**16.1 Анализ задачи о восьми ферзях**

Задача о восьми ферзях на шахматной доске 8x8 является классическим примером головоломки, требующей размещения восьми ферзей так, чтобы они не угрожали друг другу. Это означает, что ни один из ферзей не должен находиться на одной вертикали, горизонтали или диагонали с другим ферзем.

Для решения этой задачи существует несколько методов, наиболее распространенным из которых является рекурсивный алгоритм «Backtracking» (поиск с возвратом). Последовательность шагов алгоритма:

1. Определение стартовой позиции для первого ферзя на первой горизонтали и его размещение в одной из клеток.
2. Переход к следующей горизонтали и размещение второго ферзя, исключая позиции, находящиеся под угрозой первого.
3. Продолжение процесса размещения ферзей, переходя на следующую горизонталь и исключая угрожающие позиции.
4. Если на текущей горизонтали не остается допустимых позиций для размещения ферзя, необходимо вернуться на предыдущую горизонталь и изменить позицию последнего размещенного ферзя.
5. Повторение шагов 3 и 4 до нахождения решения или исчерпания всех возможностей.

Таким образом, задача о восьми ферзях иллюстрирует применение рекурсивных алгоритмов и методов поиска с возвратом в решении комбинаторных задач [18].

**16.2 Решение задачи о восьми ферзях с помощью алгоритма Backtracking**

Алгоритм «Backtracking» обеспечивает нахождение всех возможных решений задачи о восьми ферзях на шахматной доске 8x8. Если требуется лишь одно решение, алгоритм можно остановить после его нахождения.

Процесс решения начинается с размещения первого ферзя на позиции A[1][1]. После этого отмечаются крестиками клетки, которые находятся под угрозой – по вертикали, горизонтали и диагоналям (рисунок 16.1).

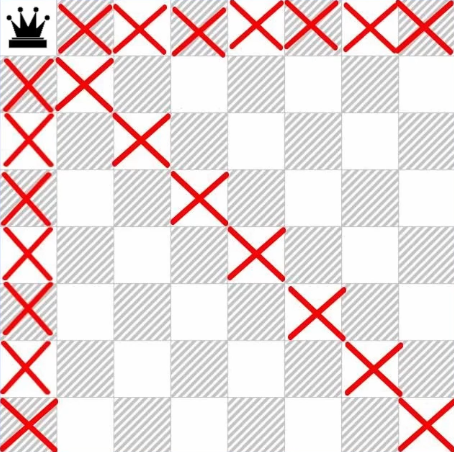


Рисунок 16.1 – Позиции, попадающие под удар ферзя

Затем второй ферзь устанавливается на первую допустимую клетку второй строки, и также отмечаются угрожающие позиции (Рисунок 16.2).

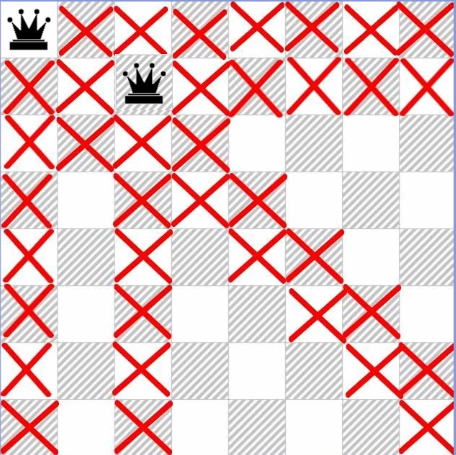


Рисунок 16.2 – Позиции, попадающие под удар второго ферзя

Этот процесс продолжается до тех пор, пока не будут расставлены все восемь ферзей или не останется доступных клеток для следующего (рисунок 16.3). Если, например, на пятой позиции невозможно разместить шестого ферзя, алгоритм возвращается к предыдущему шагу и пробует другую позицию для последнего размещенного ферзя. Если и это не приводит к успеху, продолжается возврат к предыдущим ферзя.

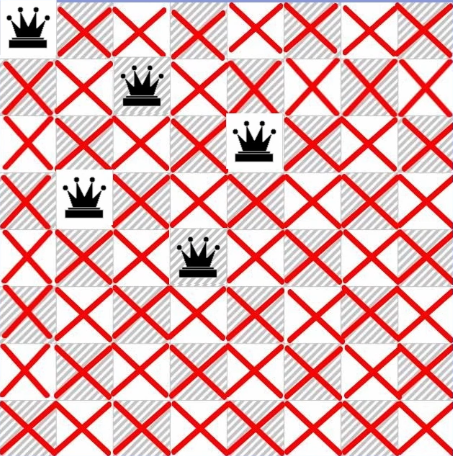


Рисунок 16.3 – Все клетки попадают под удар ферзей

Таким образом, алгоритм повторяет эти шаги до тех пор, пока не будут расставлены все восемь ферзей на доске.

**16.3 Код решения задачи о восьми ферзях**

В разделе 16.3 рассмотрен код программы на алгоритмическом языке высокого уровня С++, и предложено решение задачи о восьми ферзях на шахматной доске размером 8x8 [6]. Основные компоненты программы показаны на рисунке 16.4 и включают константу size\_arr, определяющую размер доски, и двумерный массив arr, представляющий размещение ферзей. Переменная attempt отслеживает количество попыток размещения.

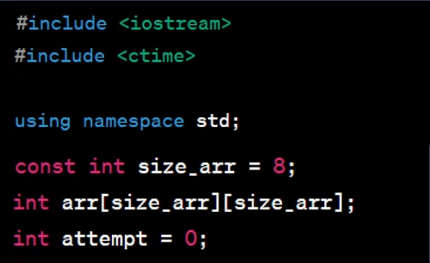


Рисунок 16.4 – Фрагмент кода программы, иллюстрирующий основные компоненты программы

Код включает несколько ключевых функций. Функция show() отвечает за вывод текущего расположения ферзей, используя вложенные циклы для отображения клеток с ферзями и пустых клеток (Рисунок 16.5).

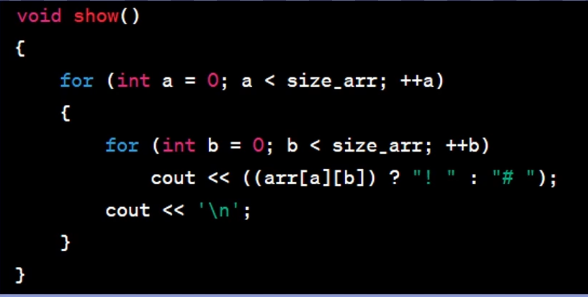


Рисунок 16.5 – Функция вывода текущих координат ферзей

Функция check() проверяет возможность размещения ферзя на заданной позиции, анализируя наличие других ферзей на вертикалях, горизонталях и диагоналях (Рисунок 16.6).

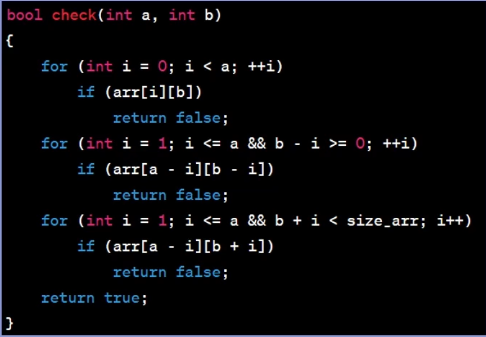


Рисунок 16.6 – Функция, проверяющая возможность размещения ферзя

Основная логика решения реализована в рекурсивной функции set(), которая принимает текущую горизонталь (строку) в качестве аргумента. Если все ферзи успешно размещены, функция вызывает show() для отображения решения. В противном случае, она перебирает все возможные позиции в текущей строке и проверяет возможность размещения ферзя с помощью check(). После успешного размещения функция вызывается рекурсивно для следующей строки, а затем ферзь снимается с доски для поиска других решений [21].

В функции main() запускается процесс размещения ферзей, вызывая set(0). По завершении работы программы выводятся все возможные решения задачи о восьми ферзях.

Таким образом, данный код эффективно находит и отображает все решения задачи о восьми ферзях на шахматной доске 8x8.