Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы» направление подготовки: 09.03.04 – «Программная инженерия»

**Лабораторная работа по теме**

**«8 ферзей»**

Выполнил студент гр. РИС-24-2б

Букач Владимир Олегович

Проверил:

Доц. Каф. ИТАС

Ольга Андреевна Полякова

(оценка) (подпись)

(дата)

г. Пермь, 2025

**Анализ задачи**

#### **1.**

* Создаётся массив queen\_positions[BOARD\_SIZE], где индекс элемента соответствует номеру строки, а значение — номеру столбца, в котором стоит ферзь.
* Пример: queen\_positions[2] = 5 означает, что в 3-й строке ферзь стоит в 6-м столбце.

#### **2.**

Для текущей строки current\_row и столбца check\_col проверяется:

1. **Столбцы**: Нет ферзей в том же столбце (queen\_positions[prev\_row] == check\_col).
2. **Диагонали**: Разница между столбцами и строками предыдущих ферзей и текущей позицией не одинакова:

cpp

Copy

abs(queen\_positions[prev\_row] - check\_col) == (current\_row - prev\_row)

Это условие проверяет нахождение на одной диагонали.

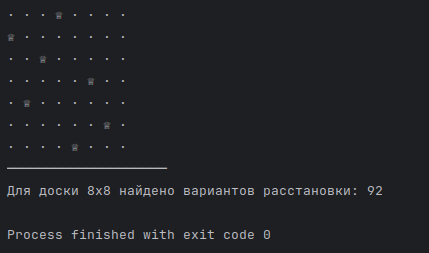
#### **3.**

1. **Базовый случай**: Если current\_row == BOARD\_SIZE, все ферзи расставлены корректно:
   * Увеличивается счётчик решений (solution\_count++).
   * Выводится текущая расстановка (displayChessBoard).
2. **Рекурсивный шаг**:
   * Для каждого столбца col в текущей строке:
     + Если позиция безопасна (checkQueenPosition), ферзь размещается в queen\_positions[current\_row] = col.
     + Рекурсивный вызов для следующей строки: findQueenSolutions(..., current\_row + 1, ...).
   * Если ни один столбец не подходит, происходит возврат к предыдущей строке.

**Код задачи**



**Пример вывода**



**Блок схема**

