```cpp
#include <iostream>
#include <vector>

void findKey(std::vector<std::vector<int>> A, int key) {
    int n = A.size();

    int row = 0;
    int col = 0;
    for (int i = 0; i < 2 * n; ++i) {
        if (A[row][col] == key) {
            std::cout << "Result: " << row << ", " << col << '\n';
            return;
        }
        if (A[row][col] > key) {
            ++row;
        } else {
            ++col;
        }
    }
    std::cout << "There isn't " << key << " in the matrix" << '\n';
}
```

затраты	итераций
C_1	1
C_2	1
C_3	1
C_4	$2n$
C_5	$2n-1$
C_6	$2n-1$
C_7	$2n-1$
C_8	$2n-1$
C_9	$2n-1$
C_{10}	$2n-1$
C_{11}	1

1. Начинаем поиск key слева вверху матрицы (элемент с индексом A[0][0]). Для индекса заводим переменные row и col для номера строки и столбца.

Идём в цикле. Если key == A[row][col], то выводим ответ.

Если key < A[row][col], то спускаемся вниз (row += 1), т.к. справа будут только элементы >A[row][col], а следовательно и больше key. Аналогично, если key > A[row][col], то идём вправо (col += 1), т.к. внизу будут элементы < key.

До самого дальнего элемента (с индексом [n-1][n-1]) необходимо 2n шагов, так как матрица размера n×n.

$$n\left\{\overbrace{\begin{pmatrix} a & b & \dots & c \\ x & \ddots & & \vdots \\ \vdots & & \ddots & \\ z & \dots & & y \end{pmatrix}}^{n}\right.$$

Вниз n шагов

Вправо n шагов

В сумме: $n+n=2n$ шагов (худший случай)

Если после прохода по циклу key не найден, выводим сообщение об этом.

2. $T(n) = c_1 \cdot 1 + c_2 \cdot 1 + c_3 \cdot 2n + (2n-1) \cdot (c_4 + c_5 + c_6 + c_7 + c_8 + c_9 + c_{10}) + 1 \cdot c_{11} =$

$= (c_1 + c_2 - c_4 - c_5 - c_6 - c_7 - c_8 - c_9 - c_{10} - c_{11}) +$

$+ (c_3 + c_4 + c_5 + c_6 + c_7 + c_8 + c_9 + c_{10}) \cdot 2n =$

$$= \left[\begin{array}{l} c_1 + c_2 - c_4 - c_5 - c_6 - c_7 - c_8 - c_9 - c_{10} - c_{11} = C_1 \\ 2(c_3 + c_4 + c_5 + c_6 + c_7 + c_8 + c_9 + c_{10}) = C_2 \end{array} \right] = C_1 + C_2 \cdot n$$

C_1 и C_2 — некоторые константы

$T(n) = O(n)$

Покажем, что $T(n) = O(n)$. Необходимо найти такие C и n, что

$C_1 + C_2 n \leq Cn$

Пусть $n_0 = 1$ и $\forall n \geq n_0$, тогд $C = \max(C_1, C_2) + 2024$.

Следовательно, $T(n) = O(n)$.