

```

#include <iostream>
#include <vector>
// #include <map>

void johnson2m(std::vector<std::vector<int>> &matrix) {
    int min_element[2];
    int indexes[2];
    int stage = 0;

    // while (stage < int(matrix[0].size() / 2));
    for (int i = 0; i < 2; i++) {
        min_element[i] = *std::min_element(std::next(matrix[i].cbegin(),
stage), std::next(matrix[i].cend(), -stage));
    }

    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < matrix[i].size(); j++) {
            if (matrix[i][j] == min_element[i]) {
                indexes[i] = j;
                break;
            }
        }
    }

    auto item = matrix[0][min_element[0]];
    matrix[0].erase(std::next(matrix[0].cbegin(), min_element[0]));
    matrix[0].insert(std::next(matrix[0].cbegin(), stage), item);
    item = matrix[1][min_element[0]];
    matrix[1].erase(std::next(matrix[1].cbegin(), min_element[0]));
    matrix[1].insert(std::next(matrix[1].cbegin(), stage), item);

    printf("END");
}

int main() {
    std::vector<std::vector<int>> m1 = {
        {9, 6, 8, 7, 12, 3},
        {7, 3, 5, 10, 4, 7}
    };

    johnson2m(m1);

    return 0;
}

```