Індивідуальна робота 4
Завірюха Еліна, МП-21

```
#include <iostream>
#include <vector>
#include <queue>
#include <unordered_set>

using namespace std;

int minMutation(string startGene, string endGene, vector<string>& bank) {
    // Convert bank to an unordered set for faster lookup
    unordered_set<string> bankSet(bank.begin(), bank.end());

    // Check if endGene is not in the bank, return -1
    if (bankSet.find(endGene) == bankSet.end()) {
        return -1;
    }

    // Define possible mutations for each character
    char mutations[] = {'A', 'C', 'G', 'T'};

    // Initialize queue for BFS
    queue<pair<string, int>> q;
    q.push({startGene, 0});

    while (!q.empty()) {
        string currentGene = q.front().first;
        int mutationsCount = q.front().second;
        q.pop();

        // Check if we reached the endGene
        if (currentGene == endGene) {
            return mutationsCount;
        }

        // Try all possible mutations
        for (int i = 0; i < 8; ++i) {
            char originalChar = currentGene[i];

            for (char mutation : mutations) {
                if (mutation != originalChar) {
                    currentGene[i] = mutation;

                    // Check if the mutated gene is in the bank
                    if (bankSet.find(currentGene) != bankSet.end()) {
                        q.push({currentGene, mutationsCount + 1});
                        bankSet.erase(currentGene);  // Mark as visited to avoid duplicates
                    }
                }
            }

            // Revert the gene back to the original state for the next mutation
            currentGene[i] = originalChar;
```

```
        }
    }

    return -1;  // No valid mutation found
}

int main() {
    // Example 1
    string startGene1 = "AACCGGTT";
    string endGene1 = "AACCGGTA";
    vector<string> bank1 = {"AACCGGTA"};
    cout << "Example 1: " << minMutation(startGene1, endGene1, bank1) << endl;

    // Example 2
    string startGene2 = "AACCGGTT";
    string endGene2 = "AAACGGTA";
    vector<string> bank2 = {"AACCGGTA", "AACCGCTA", "AAACGGTA"};
    cout << "Example 2: " << minMutation(startGene2, endGene2, bank2) << endl;

    return 0;
}
```