

Задача.

Алгоритм поиска пути очень прост. Вначале мы запускаем обычного Дейкстры. Далее мы начинаем идти из финальной точки к начальной следующим образом: каждый раз мы выбираем соседнюю вершину, до которой быстрее всего добираться из начальной. Делаем так пока не дойдем до начальной, записывая вершины, которые мы прошли.

После этого осталось перевернуть полученную последовательность, так как мы хотим путь от начала до конца.

```
std::vector<Vertex*> find_path(Vertex * start, Vertex * finish, Graph& g) {
    dijkstra_search(g, start);
    std::vector<Vertex*> result;
    Vertex * current = finish;
    while(current->value > 0) {
        result.push_back(current);
        current = (*std::min_element(current->neighbours.begin(), current-
>neighbours.end(), [](Edge * a, Edge * b){
            return a->vertex->value < b->vertex->value;
        }))->vertex;
    }
    result.push_back(current);
    std::reverse(result.begin(), result.end());
    return result;
}
```