Работа на алгоритмите използвани в приложението:

**1. Динамично програмиране (Dynamic Programming).**

Когато даден проблем е възможно да бъде разбит на по-малки, един от възможните подходи е именно „Динамичното програмиране“. По-простите проблеми водят, чрез рекурсивен подход, по-лесно до решението на дадената задача. Интегрираният алгоритъм в програмата се основава на римския принцип „Разделяй и владей“ (Bottom up). Задачата се разбива на по-малки задачи, които от своя страна се разбиват на още по-малки и т.н. докато не се достигнат достатъчно прости задачи, които да могат да се решат директно. Единственото условие е никои две подзадачи да не се пресичат.

За да сме сигурни, че сме получили оптимално решение трябва да обходим дървото напълно. За всяко ниво в дървото на решенията на задачата алгоритъмът разглежда всеки възел и съседите му и решава кой път да поеме, така че решението да е най-оптимално.

**2. Търсене с връщане (Backtracking).**

За разлика от „Динамичното програмиране“, „търсенето с връщане“ е похват, при който е възможно да се намерят всички решения на дадения проблем освен най-оптималния. Решението на задачата при този алгоритъм се конструира последователно. На всяка стъпка се прави опит за раширяване на текущото решение чрез всевъзможни налични продължения. Ако нито едно от тези разширения не доведе по-късно до пълно решение, случаят се обявява за безперспективен и алгоритъмът се връща една стъпка назад. Дървото на решенията се обхожда докато не се намери валидно решение.

**3. Метод на разклоненията и границите (Branch & Bound).**

Този алгоритъм е частен случай на „търсенето с връщане“. Преди този алгоритъм да поеме по даден клон, той проверява дали текущият изминат път няма да надвиши досегашният намерен минимален път. В такъв случай разширението спира, тъй като със сигурност няма да доведе до по-добро решние.

**4. Евристичен aлгоритъм със случайно рестартиране (Greedy Hill Climbing Steepest Ascent Random Reset).**

Този алгоритъм не гарантира, че ще намери най-доброто решение на проблема, но гарантира че ще предостави задоволително такова за сравнително кратко време. Тук се използват случайно генерирани решения, които се оценяват и на база на които се гони по-добро решение. Рестартирането на алгоритъма се използва, за да се предотврати замръзването на програмата в локален минимум на решенията на нашата задача.