Логика

Используемые принципы и алгоритмы:

1. Принцип ледокола (не пройти, пока нет первого пути/трещины во льду)
2. Алгоритм Дейкстры (все пытаются идти кратчайшим по времени путем)
3. Принцип толпы/животных в лесу (все идут по проторенным тропкам, пока для этого хватает места)
4. Использование изображения разной степени детализации (пиксельный подход) для матрицы допустимых скоростей движения

Процедура:

1. Задаем основные параметры –

d\_min - минимальное расстояние от пешеходных дорожек до зданий,

скорости движения пешеходов ([-1] – движение невозможно, [V\_0] – скорость на пустыре, [V\_N] – скорость по дорогам разных классов (N) (по линиям), [Vh] – скорость внутри зданий (внутри полигонов), [Vps] – скорость по обочинам дорог (по краям линий),

r\_peak - % населения от максимального для расчета движения в час пик,

t\_max - максимально допустимое время пути по пешеходной дорожки без использования ОТ,

p\_max - максимальная пропускная способность стандартной дорожки,

k – коэффициент снижения скорости при превышении максимальной пропускной способности дорожки

1. Создаем матрицы скоростей движения из исходных данных для каждого слоя
2. Создаем списки начальных (дома) и конечных (остановки, детсады, школы) точек движения с ограничением для района
3. Для списка координат домов создаем словари количества пешеходов для разного возраста (т.е. конечной точки назначения – дошкольники, школьники, пенсионеры, взрослые, пользующиеся ОТ
4. Суммируем матрицы и создаем общую матрицу скоростей пешеходов для карты района и пустую матрицу передвижений
5. Создаем матрицу времени (делим размер пикселя на скоорость, в зависимости от направления – диагональное движение или нет)
6. Для каждой выбранной конечной точки (садик, школа) проводим расчет минимального времени движения по пикселям карты/матрице скоростей от каждого из домов (в произвольном порядке из списка строящихся домов) в два этапа – первый этап в условиях отсутствия дорожки от дома и уже имеющихся дорожек к ранее обсчитываемым домам, второй этап уже с учетом проложенной на первом этапе дорожки от обсчитываемого дома (алгоритм Дейкстры). На основе координат путей и количеств пешеходов создаем матрицу путей движения (для точек на пути дорожек получаем общее количество передвигающихся пешеходов). При превышении p\_max для каких-то точек на матрице передвижений уменьшаем допустимую скорость движения в этих точках на коэффициент k и продолжаем расчеты кратчайших по времени путей
7. Для всех итераций по п.7 проводим расчет целевых функций – сумма произведений времени пути между конечными/начальными точками и количества пешеходов для каждого дома
8. Итерации по п.п. 7-8 производим многократно и находим минимальное значение целевой функции, и, соответственно, набор координат пешеходных дорожек
9. Выводим оптимизированную матрицу пешеходных дорожек в виде изображения

Дополнительно – можно ввести параметр максимальных капзатрат/бюджета (максимальной длины дорожек на кв. км) и при превышеини параметра для сети дорожек проводить перерасчет с учетом ограничения