1. 初始化种群、编码

由于将坐标表现为基因的形式，而起始坐标和终点坐标不可分割到两个染色体上（一条染色体视为一辆出租车，因为本次问题出租车仅使用一次，即600多条染色体）。所以，初始坐标和终点坐标不可分割？

而基因仅具代表作用，所以将其化为二进制形式（2的10次方即为1024，11次方即为2048，足够大）。总共1001位乘客，一位乘客两个坐标，2002<2048，即11位二进制，每条染色体8个11位二进制，总共600多条染色体，组成初始种群。生成一个随机的三维矩阵即可。

二、选择choose

轮盘赌。本算法设计的适应度与ans成反比，适应度越高，ans越低，所以选择的时候是反向思考，采取淘汰法，选中者被淘汰，进行变异等操作。由于乘客不可舍弃，所以不能舍弃某个父代，所以只能对适应度高的进行保留，低的进行变异。（不知道禁忌搜索算法是不是这个意思）

1. 交叉

可染色体多位对多位的交叉，实际意义即乘客被另外的出租车运走。此时注意初始坐标和终点坐标不可分割，同时交换。

1. 变异

上下车顺序的变化，某乘客下车不得在另外乘客上车前，否则该出租车载客不止一次。

还有将一队二进制分割成两队，实际意义即为分成两辆出租车载客。

1. 适应度计算

一个因素是减少乘客等待时间，即所有出租车行程最短，此为首要因素。

另一个是增加空乘率，即使载客的出租车尽量满载。

适应度按权重计算。

笔者认为可以这么搞，如果汽车一小时跑60公里，那么一分钟一公里，所以一个乘客等一分钟就相当于汽车多跑一公里。假如车可以满载，那每空出一辆车就相当于四名乘客，而经过计算，一名乘客平均有3.95单位坐标（500米）的行程，四名就是7.9公里。所以适应度计算公式为 ans=总路程+载客车辆\*7.9（单位，公里） ，ans越高适应度越低。（总里程别忘了考虑出租车到第一名乘客的距离）

一辆车的适应度应该是 ans=载人路程+缺人数\*1.975.