输入n，生成一个n阶关于对角线对称的矩阵代表节点为n的图，比如（2，3）的值=（3，2）的值，并且主对角线（1，1）到（n，n）的值全部为0，输出这个矩阵。

算法1：初始化

1：输入种群G的大小g，输入失效结点集合F（初始F为空集，依次输入fi，输入一个回车一次，将输入的数加入F，输入为空时执行下一步）；

2：令A={},B={},T={}；

3: 遍历F,以其中的fi为根结点构造修复树；

令A={fi},B={n-F元素个数个结点},Ti={}；

if B≠{}：

取出B中的一个点b，连接到A中的任一个点a（b作为a的子结点）

把b从B中去掉并加入A，并且将b的父节点a加入Ti，直到B={}

4.得到一个种群个体Gi={T1,…,T|F|}；

6.循环执行3-4，直到|G|=g。

比如输入n=5，输出一个5阶沿对角线对称的矩阵，（2，3）的值=（3，2）的值，并且主对角线（1，1）到（n，n）的值全部为0，输出这个矩阵。

输入种群大小g

然后依次输入fi，比如2，3，然后F={2，3}

从F中按顺序先后选择，先选择2，A={2}，B={1，4，5}

依次从B选择元素，然后从A选择元素，把A选择的元素作为B选择元素的父节点，加入Ti，把B选择的元素从B去掉，加入A中，把B的父节点加入T2中

比如从A选择2，B选择1，那么2就是1的父节点，把1的父节点加入T2，T2={2}，现在A={2，1}，B={4，5}，然后接着选择，直到B={}，构造2为根节点的树，把T2加入个体Gi

从F中再选择3，直到F中所有元素都选择完

个体Gi={T2，T3}

依次生成g个Gi，加入种群G

G={G1，G2……Gg}

={{T2，T3}，{T2，T3}……{T2，T3}}

算法2：交叉：

步骤1：设交叉概率Pc=0.8

步骤2：从初始种群 G 中随机选择两个个体 parent1 和 parent2 作为交叉亲代，随机产生[0，1]之间的随机数p1，如果p1＜Pc，则随机执行步骤3或步骤4，否则步骤6；

步骤3：随机选择当前失效节点集合中一半的节点，调换这些节点在parent1和parent2中的修复拓扑。得到交叉子代crossSon1和crossSon2，将之加入交叉子代crossOffspring；

步骤4：随机选择多个失效节点fi，分别从parent1和parent2的fi修复树中随机选择一个相同的节点j，将两棵树中节点j的父节点进行调换，得到交叉子代crossSon3和crossSon4；然后执行步骤5。

步骤5：沿着节点j的父节点向上查找，若其中任意一个查找到j则重新执行步骤4，若两方都查找到节点fi则停止查找，将交叉子代crossSon3和crossSon4加入交叉子代crossOffspring；

步骤6：将parent1 和 parent2直接加入交叉子代crossOffspring。

算法3：变异

步骤1：设变异概率Pm=0.15

步骤2：从初始种群G中随机选择一个个体parent作为变异亲代，随机产生[0，1]之间的随机数p2，如果p2＜Pc，执行步骤3，否则步骤4；

步骤3：随机选择多个失效节点fi，分别将fi的修复树中的元素分别加入到空的集合Fi，对Fi进行去重，得到中间节点的集合mid，用构成树的所有节点的集合whole减去中间节点的集合得到叶子节点的集合leaf，随机选择一个叶子节点，再随机选择一个不是叶子节点父节点的中间节点，将叶子节点的父节点改为选择的中间节点得到新的修复树，分别将新的修复树替换原来的修复树最终得到变异子代 mutationSon1，加入变异子代mutationOffspring；

步骤4：将parent直接加入变异子代mutationOffspring

算法4：局部搜索

步骤1：选择个体进行局部搜索，对局部搜索个体的每条边进行复用标记，所有树没用过标记0，一棵树用过标记1，以此类推，再判断瓶颈链路是否是由复用引起的，如果是由复用引起的执行步骤2，否则执行步骤3；

步骤2：随机选择公用瓶颈链路修复树中的一棵fi，随机选择一个非瓶颈链路中节点的节点，判断该节点与瓶颈链路子节点链路增加一次复用之后可用带宽是否高于瓶颈链路带宽，若高于则删除当前链路，采用刚刚寻找到的连接，并沿着父节点向上查找确保不会出现该子节点，否则重新执行步骤3并最多随机选择n/10次，如果n/10次都没寻找到则将原个体直接加入局部搜索子代 localSearchOffspring，如果搜索到了分别用这个新的修复树替换原基因位的修复树，将更新后的个体加入局部搜索子代 localSearchOffspring

步骤3：随机选择一个非瓶颈链路中节点的节点，判断该节点与瓶颈链路子节点链路增加一次复用之后可用带宽是否高于瓶颈链路带宽，若高于则删除当前链路，采用刚刚寻找到的连接，并沿着父节点向上查找确保不会出现该子节点，否则重新执行步骤3并最多随机选择n/10次，如果n/10次都没寻找到则将原个体直接加入局部搜索子代 localSearchOffspring，如果搜索到了分别用这个新的修复树替换原基因位的修复树，将更新后的个体加入局部搜索子代 localSearchOffspring

算法5：选择

步骤 1：计算初始种群 population 与交叉子代 crossOffspring 以及变异子代 mutationOffspring 和局部搜索子代 localSearchOffspring 的并集中每个个体的适应度， 并依据适应度对集合中的个体进行升序排序；

步骤 2：截取序列中前 populationSize\*0.1 个个体作为精英个体，直接遗传至新一 代种群 populationnew 中；

步骤 3：对序列中剩余的个体进行随机选择，将结果加入 populationnew，直至新 一代种群大小等于 populationSize

综上：

步骤 1：(参数初始化)初始化种群规模参数 populationSize，交叉概率 PC 和变异概率 PM，及参数 ctime，ctraffic；

步骤 2：(种群初始化)采用算法 1 随机生成 populationSize 个个体，作为初始种群 P(0)，初始迭代次数 t=0；

步骤 3：(交叉操作)在当前种群 P(t)中随机选择两个个体作为交叉亲代，以概率 PC 执行算法 2，得到交叉子代 crossOffspring；

步骤 4：(变异操作)从当前种群 P(t)中随机挑选一个个体作为变异亲代，以概率 PM 执行算法 3：产生变异子代 mutationOffspring；

步骤5：(局部搜索)在种群P(t)∪crossOffspring∪mutationOffspring中的所有个体中选择部分， 依次执行算法 4，得到局部搜索子代 localSearchOffspring；

步 骤 6 ： ( 选 择 操 作 ) 计 算 种 群 P(t) ∪ crossOffspring ∪ mutationOffspring ∪ localSearchOffspring 中的所有个体适应度，以此执行算法 5，得到新一代种群 P(t+1)， 同时令 t=t+1；

步骤 7：(终止)如果连续G次种群中适应度函数值最低的个体为同一个体，就将种群 P(t)中适应度函数值最低的个体作为全局近似最优修复拓扑输出。否则跳至步骤 3 继续执行。