1. 使用灰狼算法和遗传算法结合
2. 对灰狼算法使用存档机制

3、在算法后期加入拥挤度反向叛变机制

4、加入遗传算法重复机制，让个体的一段发生重复复制的操作

问题：确定主副种群还是两个平行种群

使用主副种群，发现比较哪里有更有效的个体(判断)，然后把这些个体加入到主种群当中去

核心编程尝试，不写死任何，只确定存储结构，不确定存储个数，ARG需要提前进行确认

这样就可以确保在种群个体数发生变化时进行操作

按照步骤一步步实现

1. 单一多目标遗传算法 NSGA--II
2. 加入遗传重复机制
3. 加入档案机制
4. 单一灰狼算法
5. 灰狼拥挤度反向叛变
6. 灰狼多领导叛变
7. 当系统最优值持续不发生变化后，自适应修改部分个体的参数，提高他们的变化参数。 交叉率 步长。。
8. 已经实现了通过迭代次数进行操作、根据距离决定是否继续进行操作

换一种书写代码的方式，让他可以在自己的环境里自己玩。

1、强化学校 ：考虑是否可以先运行半次专门用于训练，在进行

直接使用，大幅提高计算精准度或运算时间

1. 建立引入结构并且进行

实现的是随机丢失一部分种群，防止趋同化

编写中想到的可以改进的点子

1、自适应参数 ：观察每次总适应度的进步程度 在开始阶段和进步较少的阶段，调整参数 控制重复的概率

3、在使用 重心进行操作后，可以根据不同RANK排序算出来的重心以及对他们的厌恶程度进行不同程度上的躲避。

同时为了保护最优秀的个体，专门设立一个保护前几名优秀个体的保护机制或者直接设计成后面的只推动RANK为2的个体进行进化，保证Rank为1的纯种性和独立唯一性