这周主要学习了遗传算法并且进行了相关代码的编写。

遗传算法的步骤：

1：def __init__(self, size, chrom_size, cp, mp, gen_max)初始化第一代染色体，染色体可表达为二进制数字串。

2：然后采用适应度函数 evaluate(self)分别计算每一条染色体的适应程度，并根据适应程度计算每一条染色体在下一次进化中被选中的概率。

3：通过轮盘赌选择法 select(self)进行染色体的选择来产生下一代的染色体。

4：通过“交叉” cross(self, chrom1, chrom2)，生成染色体；

5：再对交叉后生成的染色体进行“变异”操作 mutate(self, chrom)；

6：在进行完一轮遗传变异之后，用适应度函数对这些新的后代进行验证，如果函数判定它们适应度足够，那么就会用它们从总体中替代掉那些适应度不够的染色体。使用 reproduct_elitist(self)保存最佳个体。

接下来利用遗传算法来解决二元函数的最大值求解问题。

问题如下：

F(x,y)=x*x+y*y,其中-10<=x，y<=10.易知当 x,y 都取到 10 的时候可以获得最大值为 200.

其中设置种群的个体数量为 50，染色体长度为 25，交叉概率为 0.8，变异概率为 0.1,进化最大世代数为 100.

假如设定求解的精度为小数点后 6 位，可以将区间[-10,10]划分为 20*10^6 个区间，需要使用 25 个二进制数字来表示。

最后结果如图：

```
gen: 93 max: 192.6034174025968
gen: 94 max: 192.26406911850677
gen: 95 max: 192.80477308846096
gen: 96 max: 193.29542706912787
gen: 97 max: 193.26092664825808
gen: 98 max: 192.89688506861688
gen: 99 max: 194.01018071837876

Process finished with exit code 0
```