SESSDSA'20

问题解答
实习报告
基因遗传算法

数据结构与算法（Python）-21/0526

陈斌 gischen@pku.edu.cn 北京大学地球与空间科学学院

目录

SESSDSA'20

问题解答

- 1.有些在本地跑的很好的代码在天梯上就会出bug， 比如数组越界等等问题
- 2关于博弈树的优化方式
- 3大作业决策树的优化， alphabet a剪枝效果不大怎么办
- 7可不可以天梯加入人机对战测试?
- 9递归的时间复杂度怎么求?比如慕课的八皇后算法
- 6可不可以具体讲一下神经网络在对抗性游戏中如何训练(比如如何把输赢这反向传播回去)，最好有具体实现

八皇后问题时间复杂度

6 多选 (3分) 以下是使用递归算法对N皇后问题求解的不完整代码:

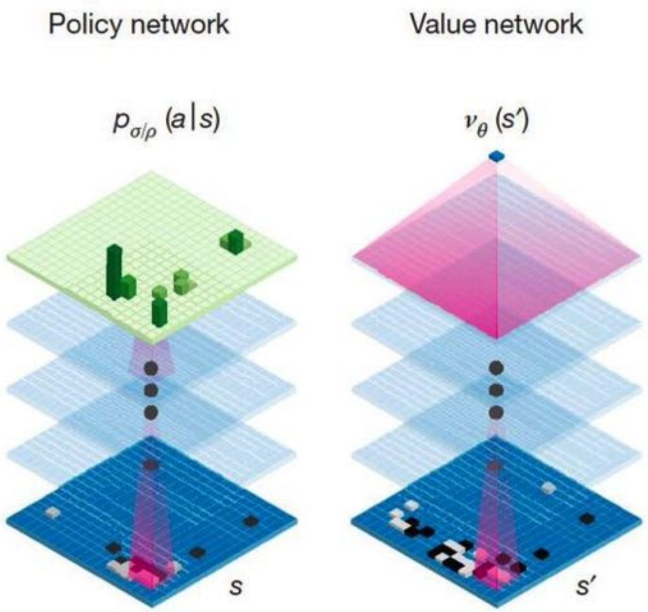
```python
def solveNQueen(N):
    pool = # <A>
    def queen(cur=0):
        if cur == len(pool):
            return # <X>
        res = # <Y>
        for col in range(len(pool)):
            pool[cur], flag = col, True
            for row in range(cur):
                if pool[row] == col or abs(col - pool[row]) == cur - row:
                    flag = False
                    break
            if flag:
                res += queen(cur+1)
        return res

    return queen(0)

# test
print(solveNQueen(8))
```

阅读代码，选出正确的选项

AlphaGO

- 结合了蒙特卡洛树搜索、估值神经网络、策略神经网络

Review: AlphaGo



- **Policy network:** initialized by supervised training on large amount of human games
- **Value network:** trained to predict outcome of game based on self-play
- Networks are used to guide Monte Carlo tree search (MCTS)

D. Silver et al., [Mastering the Game of Go with Deep Neural Networks and Tree Search](), Nature 529, January 2016

问题解答

- 4期末大报告具体要求?
- 5大作业小组账号什么时候发
- 8期末报告是什么情况?
- 10求实验报告模版

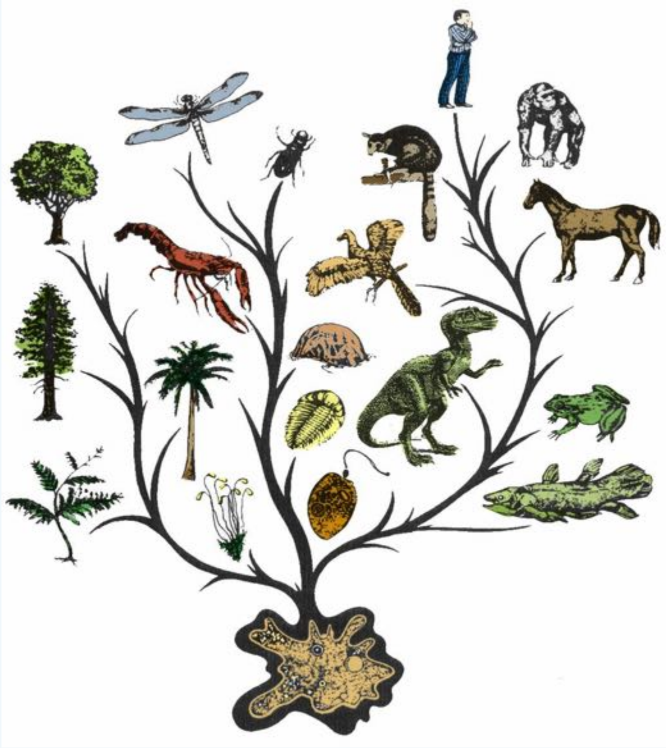
炼丹术/炼金术

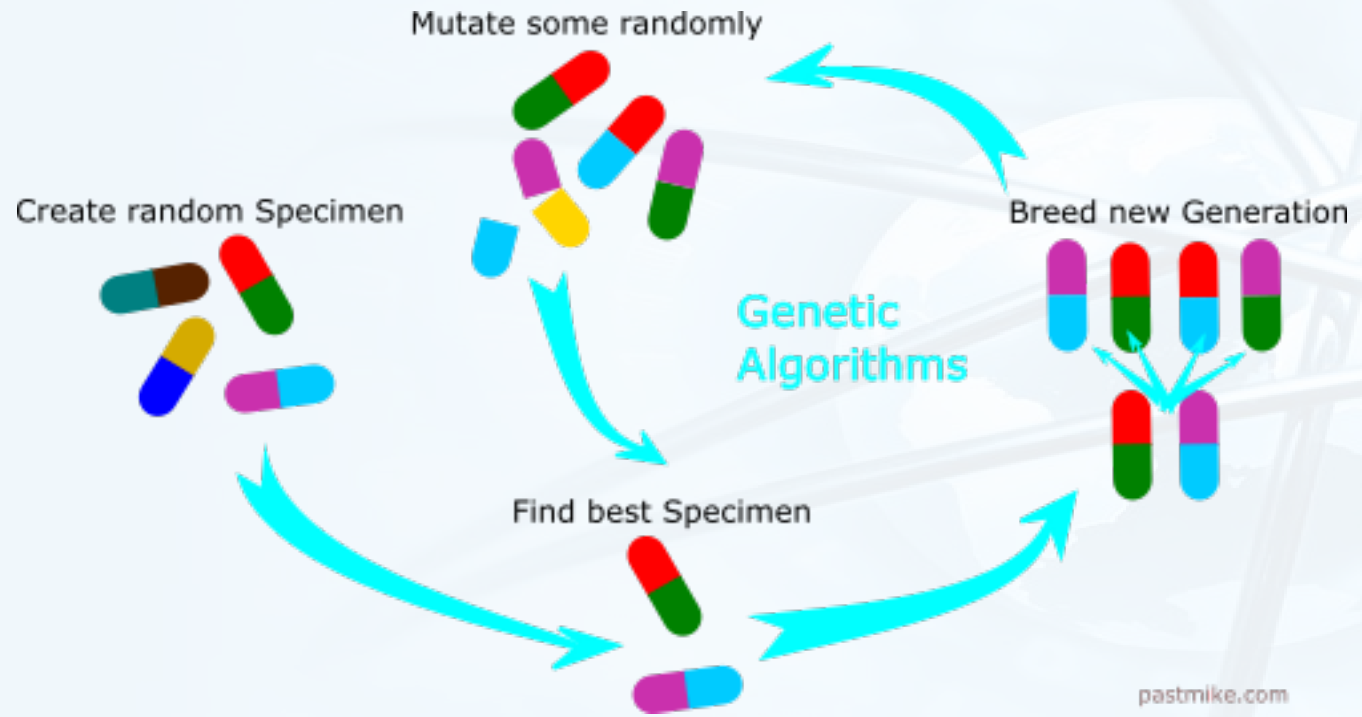
仿生算法

- 是人类模仿生物的功能与行为进而总结出来用于解决问题的方法。
- 通常通过模拟**自然生物进化**或者**群体社会行为**来进行随机搜索。
- 适用于传统算法难以解决的**大规模复杂优化**问题。

仿生算法举例

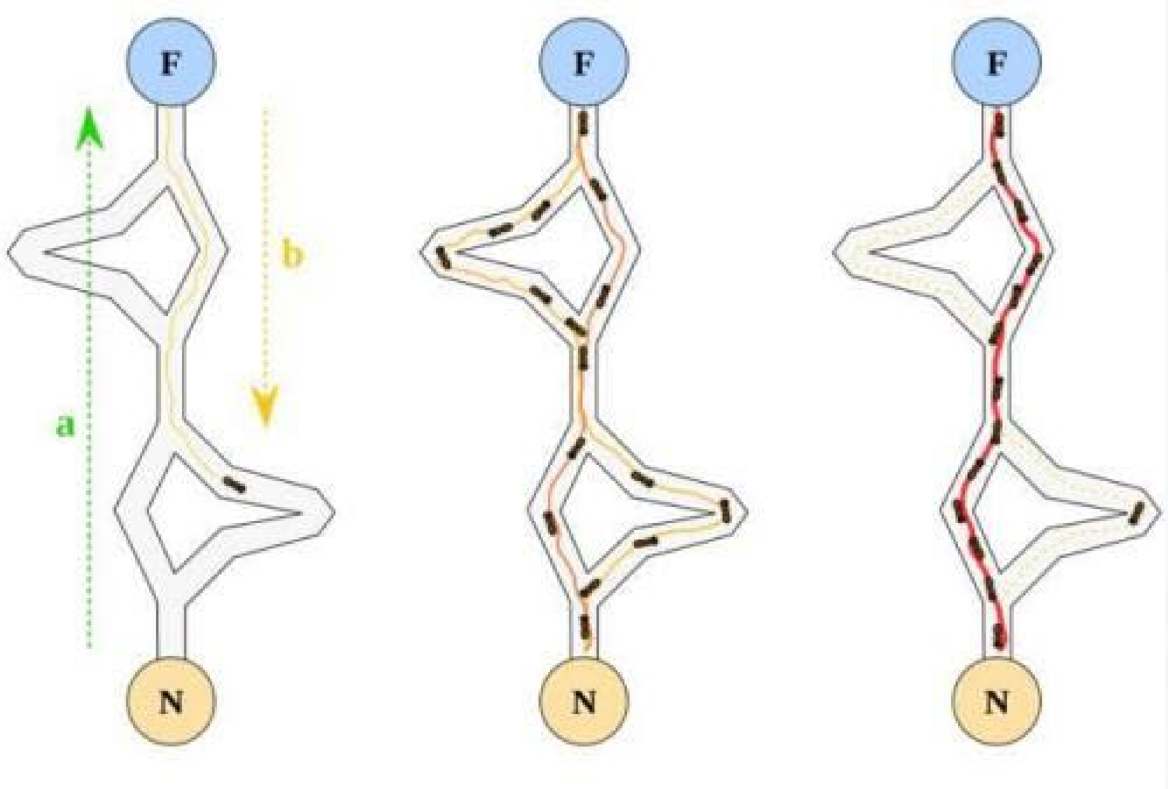
- 模仿生物进化的遗传算法。





仿生算法举例

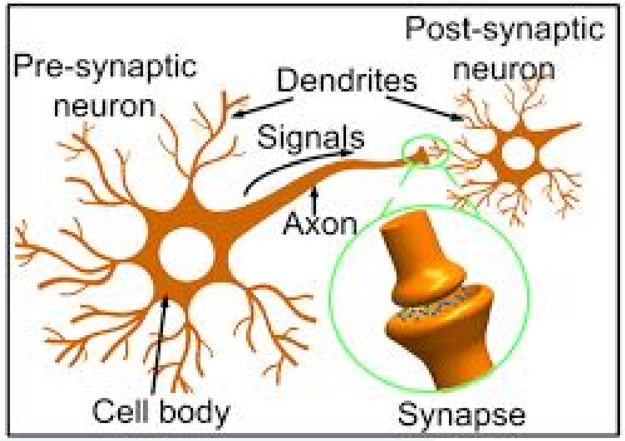
- 模仿蚂蚁寻路的蚁群算法。

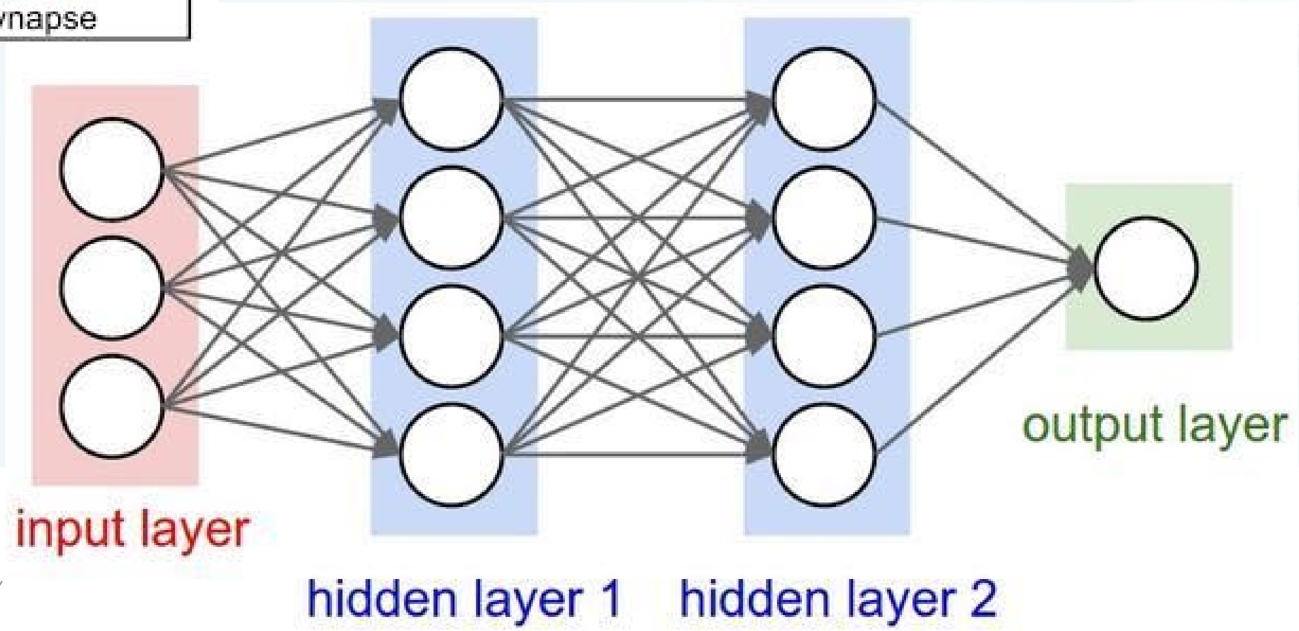


1 2 3

仿生算法举例

- 模仿神经网络系统的人工神经网络。

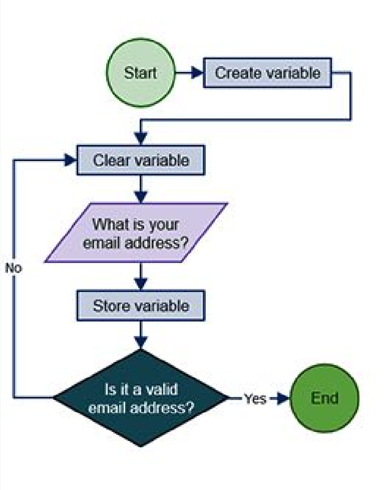


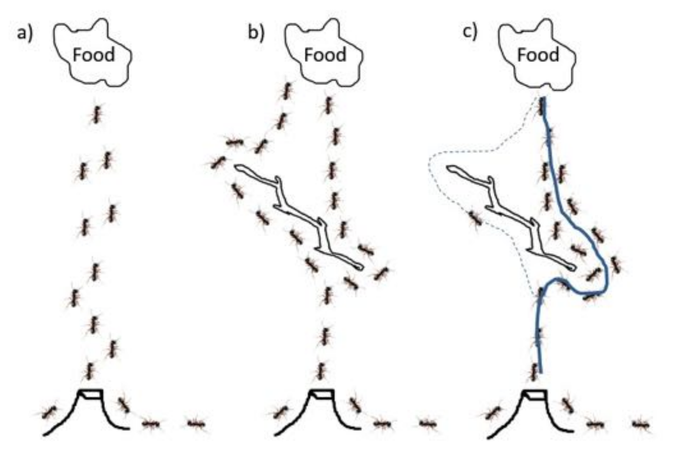


仿生算法与一般算法的区别?

一般算法

基于经验或者已知的规则，针对特定的输入能够计算出确定的输出结果





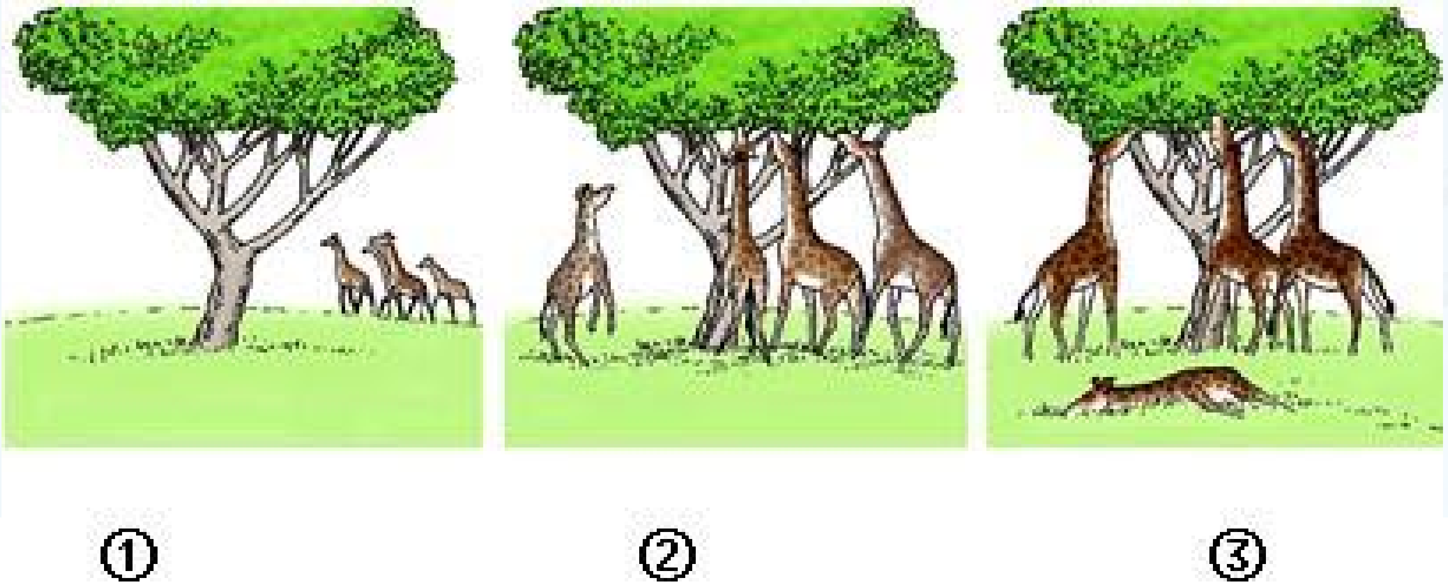
仿生算法

模仿生物的功能与行为

通过大量演变来逼近结果，演变方法是自然法则：优胜劣汰，适者生存

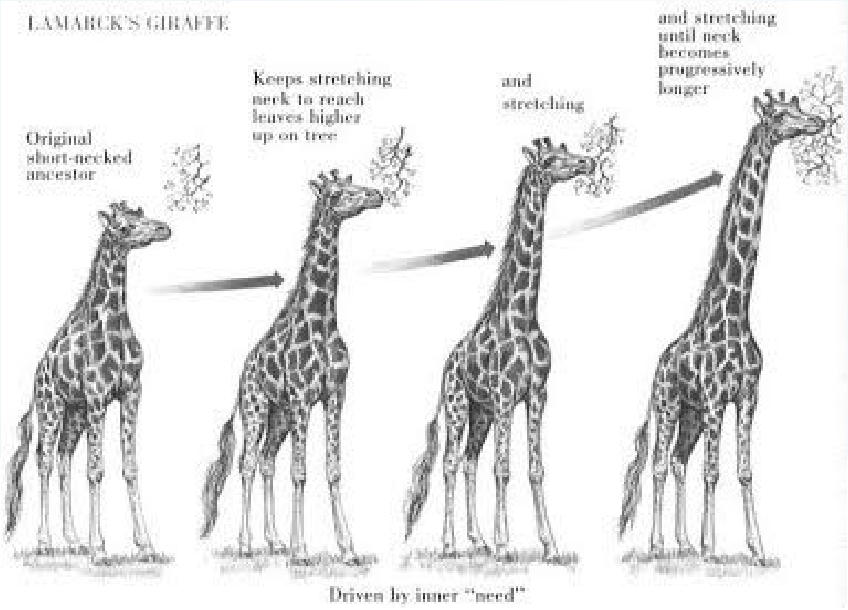
基因遗传算法

- 遗传算法是一种受达尔文自然进化理论启发的搜索启发式算法。
- 该算法反映了自然选择的过程，其中选择最适合的个体进行繁殖以产生下一代的后代。



自然选择的概念：

- 自然选择的过程始于从群体中**选择最适合的个体**。
- 如果父母有很好的适应环境，他们的后代将继承父母的特征，并继续影响下一代。
- 这个过程不断迭代，最后将找到最适合的一代。



搜索问题中的自然选择

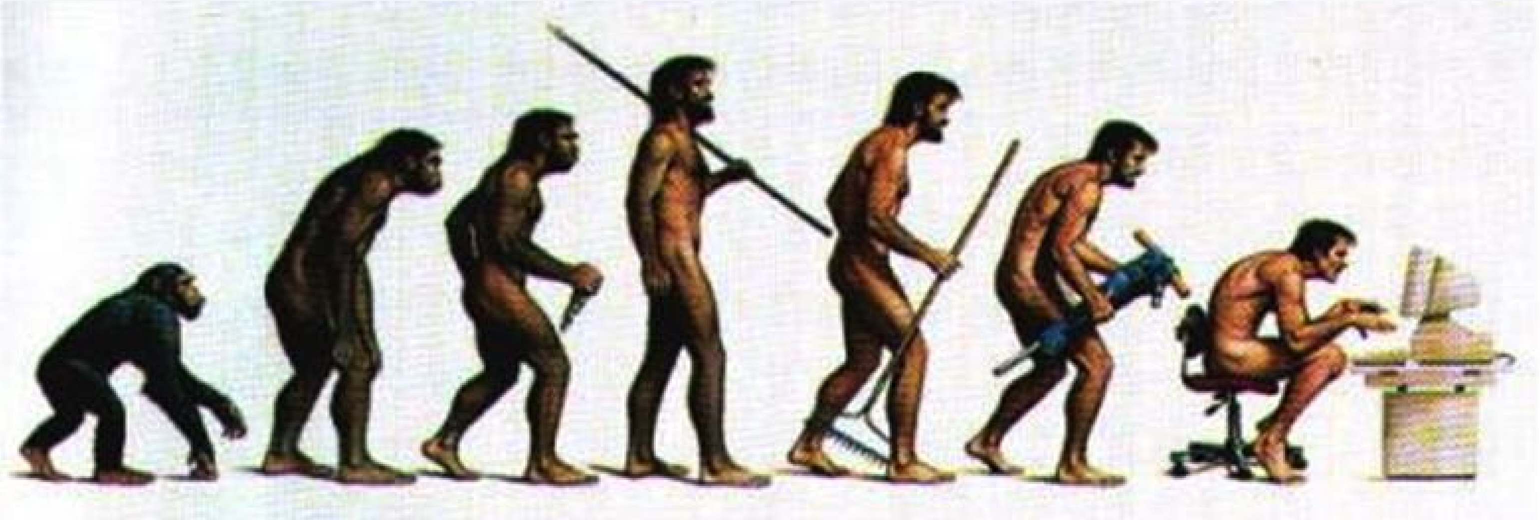
- 自然选择的概念可以应用于搜索问题。
- 我们考虑问题时，从一个一般的解决方案入手
- 不断优化，产生一组最好的解决方案。

基因遗传算法的组成：

- 编码（初始化）gene coding（population）
- 适应度函数fitness
- 选择selection
- 交叉crossover
- 变异mutation

基因遗传算法与自然界遗传的对照

- 编码（初始化）→老祖先的诞生
- 适应度函数→物种特征适应环境的判定条件
- 选择、交叉、变异→自然选择与遗传变异（突变与基因重组）



基因遗传算法的主要流程：

- 随机生成初始群体。
- 开始主循环（结束的标准可以是迭代次数，或者适应度达到某个要求）。

a） 执行策略，计算当前群体中所有个体的适应度；

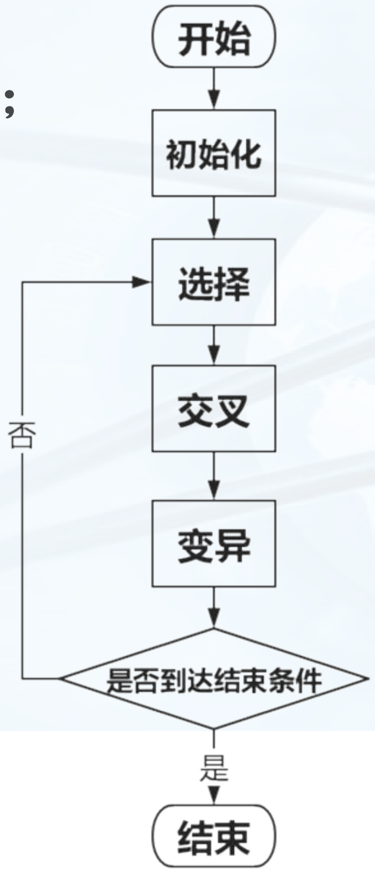
b） 从当前群体中，选择精英作为下一代的父母；

c） 将选出的精英父母配对；

d） 以极小概率将子代变异；

e） 将子代个体添加到新群体中。

适应度最高的一批



基因遗传算法的应用

函数优化、组合优化

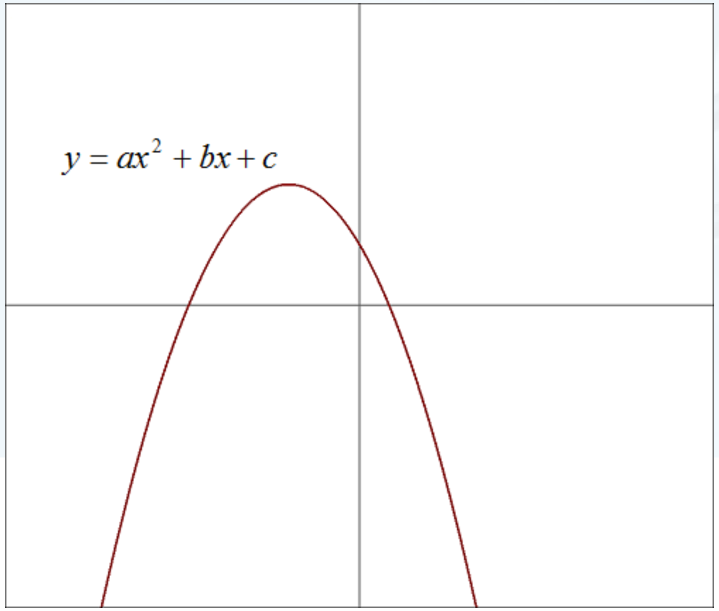
优化是通过改变参数来**最大化**或**最小化**目标函数。

所有可能解（参数值）的集合构成**搜索空间**。

在这个搜索空间中，**有一组点**是最佳解决方案，优化的目的就是在搜索空间中**找到**这样一组点。

例如左图寻找最大值

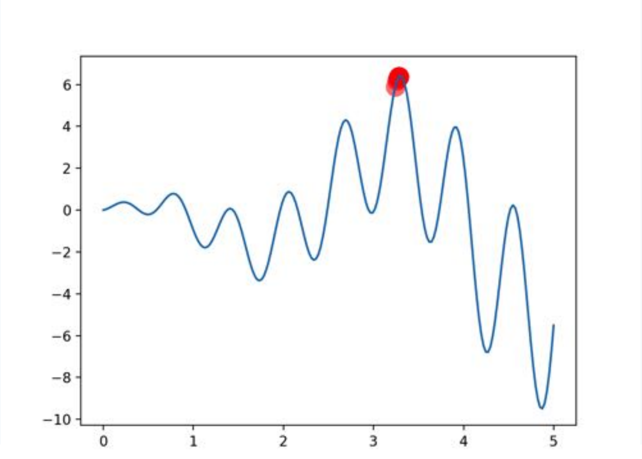
如何搜索比较快？

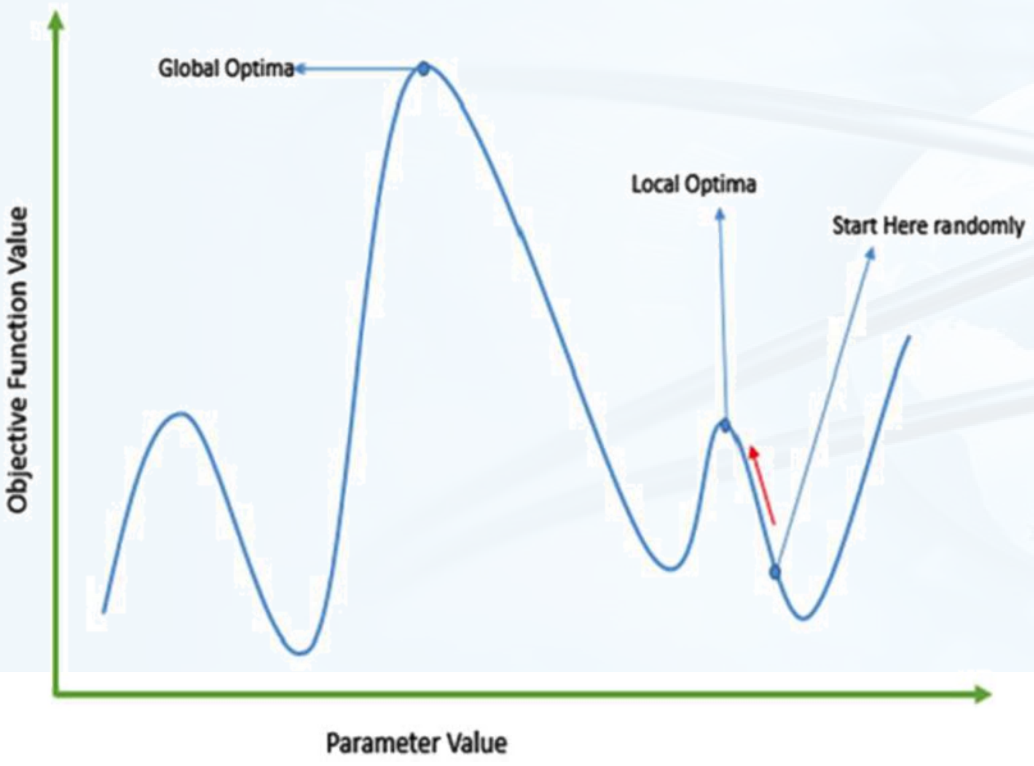


基因遗传算法的优缺点

› **那么下图的函数找最大值呢？**

用原来的搜索方法会有什么问题？



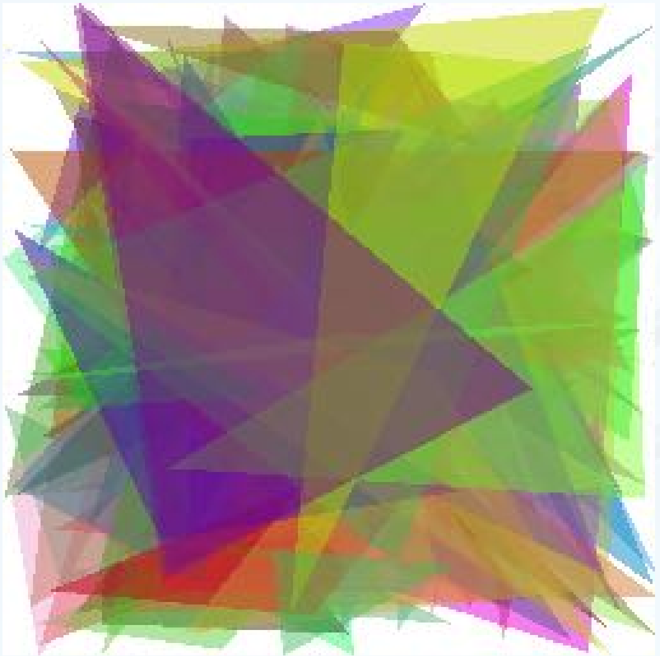


喜马拉雅山脉里随机跳跃的袋鼠

〉 找到珠穆朗玛峰

拼图游戏

- 如何用128个**半透明的彩色三角形**组合成指定的图案？
- 如果用**搜索**来解决的话，需要搜索多少三角形的组合？



拼图游戏

- **扇贝的故事：**

  海岸边有一大群扇贝

  拾海鲜者专找不像其图腾的扇贝吃

  经过漫长的年代后

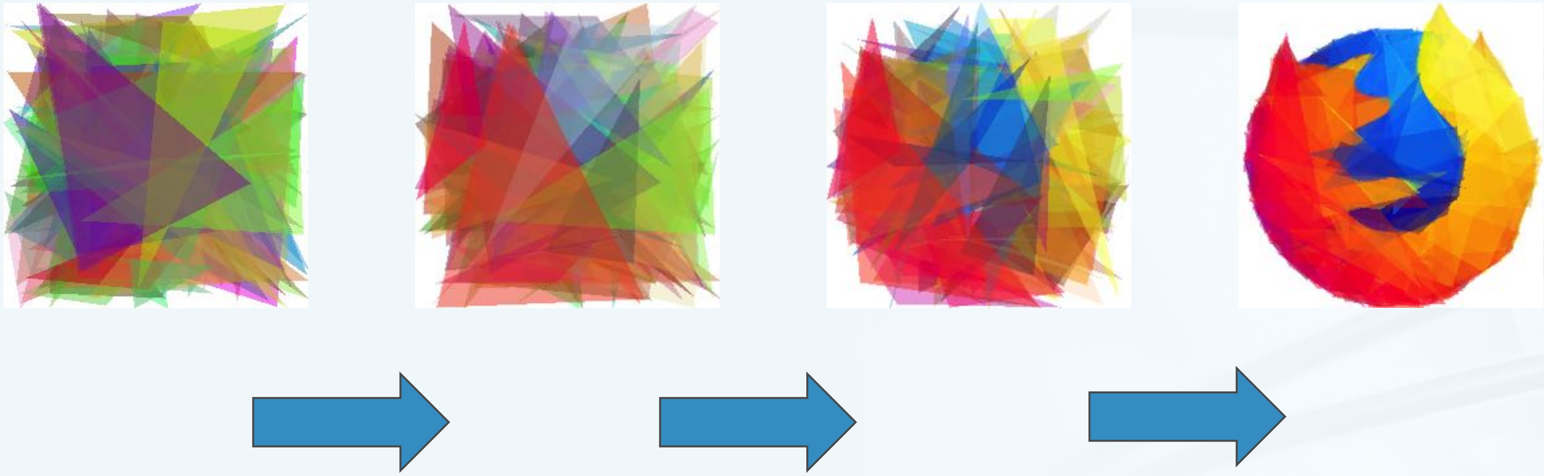
  存活的扇贝贝壳上都印着很像图腾的图案

思考

- 扇贝的故事如何启发我们解决拼图问题？

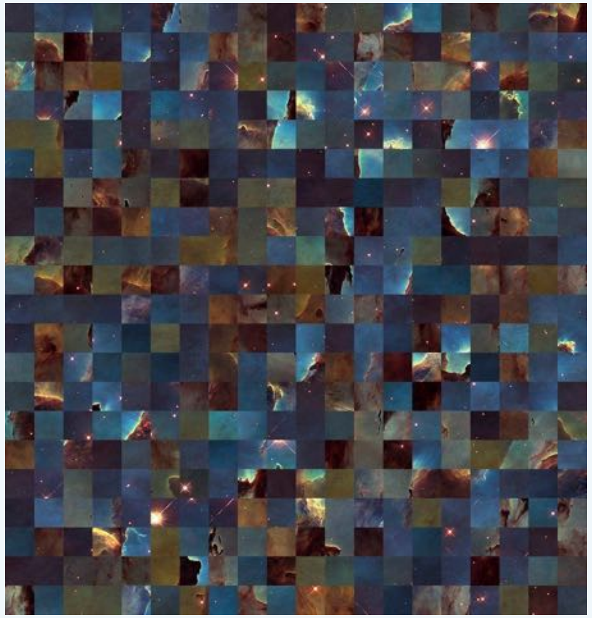
拼图游戏

〉 **先看一下成果图吧~**



拼图游戏

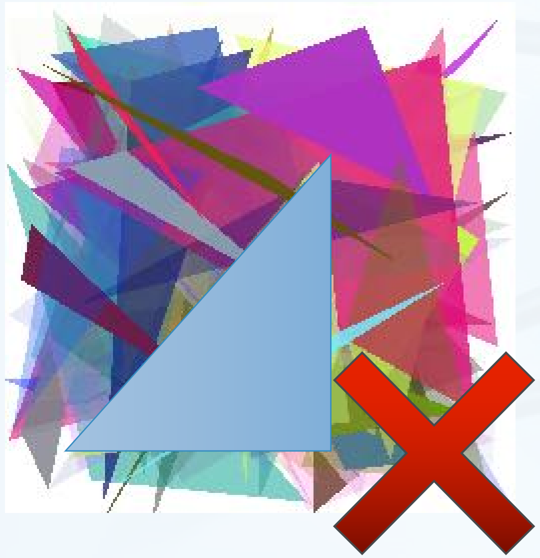
› 除了利用随机三角形生成图标，还可以利用遗传算法完成拼图



拼图游戏

原理

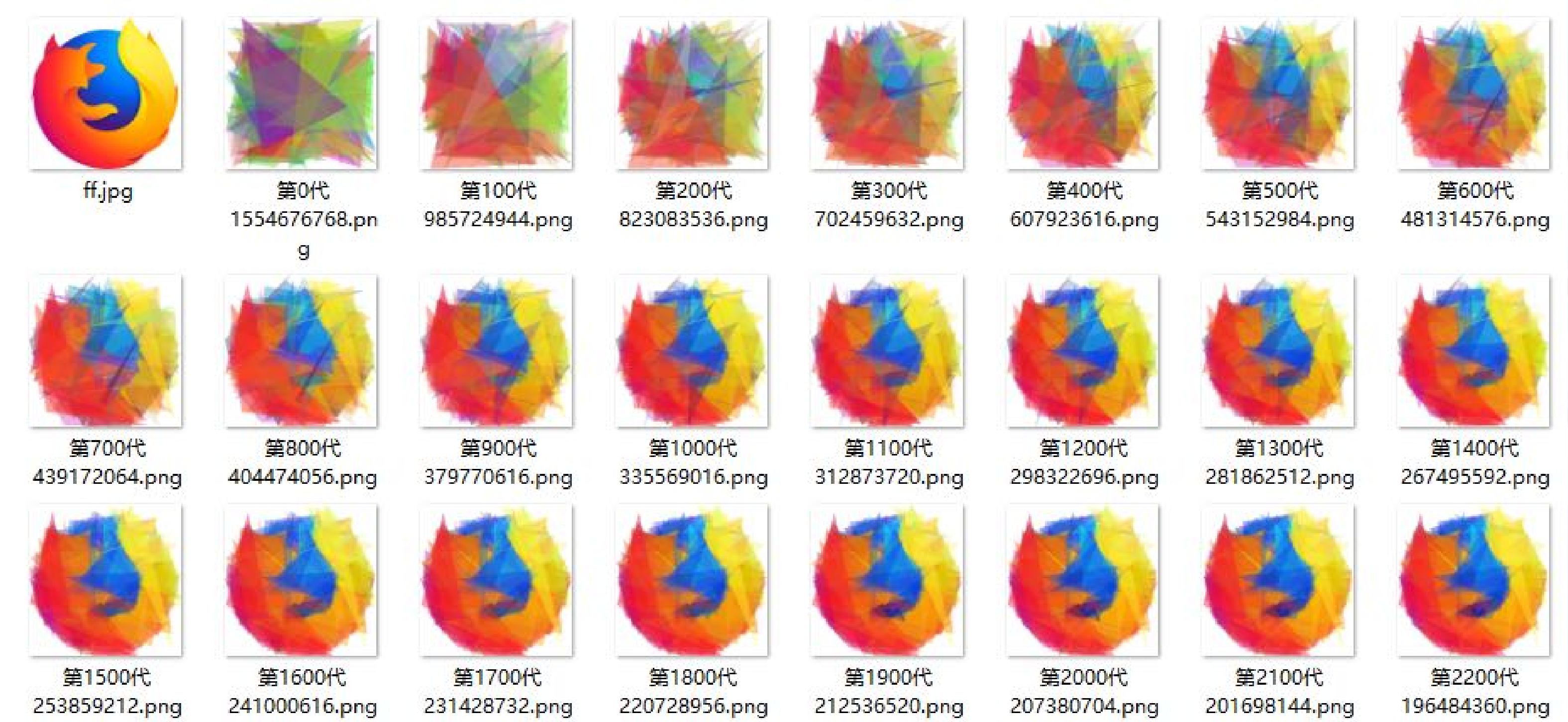
与自然界进化一样，要设立一个淘汰机制且不断提高标准，淘汰掉与样本图不像的图案



拼图游戏

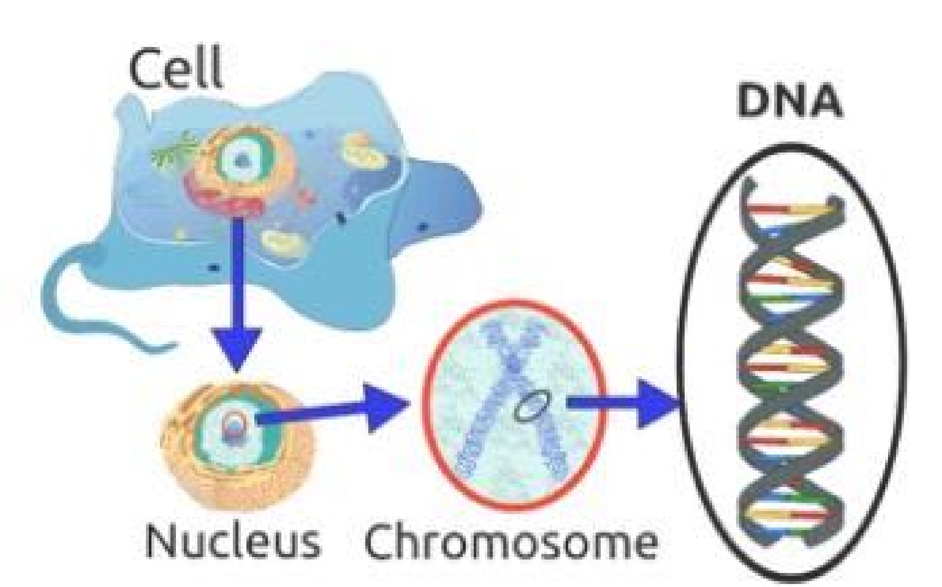
总结：

用三角形拟合图案是应用遗传算法的一个实例。覆盖基因、遗传、变异、选择等方面。



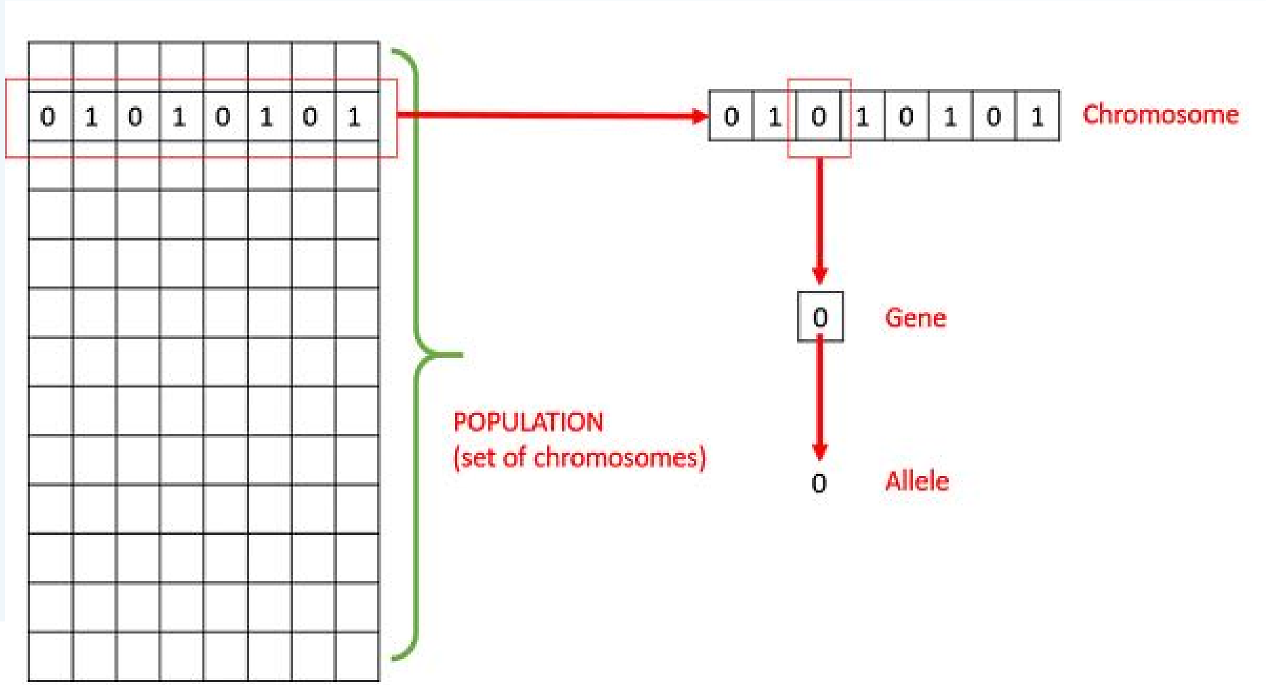
生物基因

- 基因：是遗传的基本单元
- 在一个生物的任何一个细胞中，都有着相同的一套染色体
- 染色体中决定生物各种性状的信息就是基因（DNA片段）



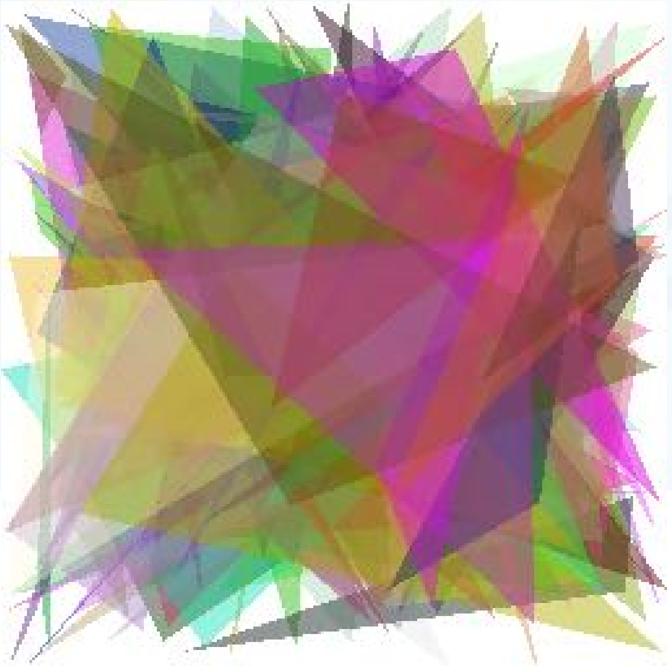
生物基因

- 若干条染色体组成一个个体，每条染色体中有若干个基因。
- 每个基因都编码了一个独特的性状。



拼图的基因

- 如果我们把128个半透明三角形组成的东西看成一个生物个体的话，它的样子是由这些三角形决定的。
- 每个三角形可以看作一个染色体
- 三角形的不同特征可以看作不同基因



三角形的特征

确定一个三角形的因素有哪些？

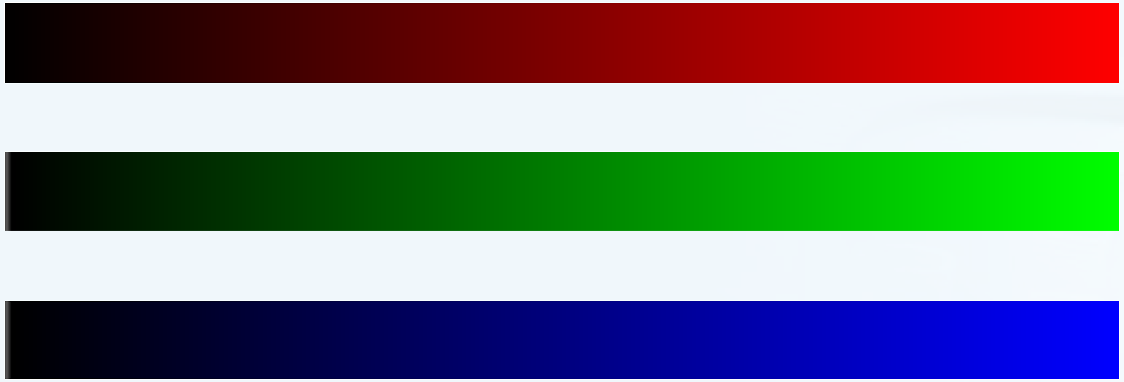
1.颜色

2.位置和大小：三个顶点坐标

3.透明度

颜色：RGB颜色模型

- 三原色（红、绿、蓝）



位置和透明度

位置

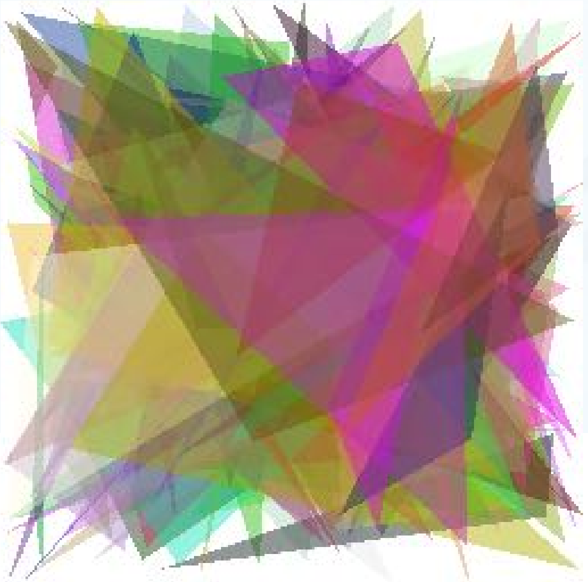
三角形的三个顶点位置坐标(x1, y1, x2, y2, x3, y3)决定了三角形的大小和位置，也决定了三角形的另一个基因。

透明度

取值在0%-100%之间，决定了三角形的第三个基因。

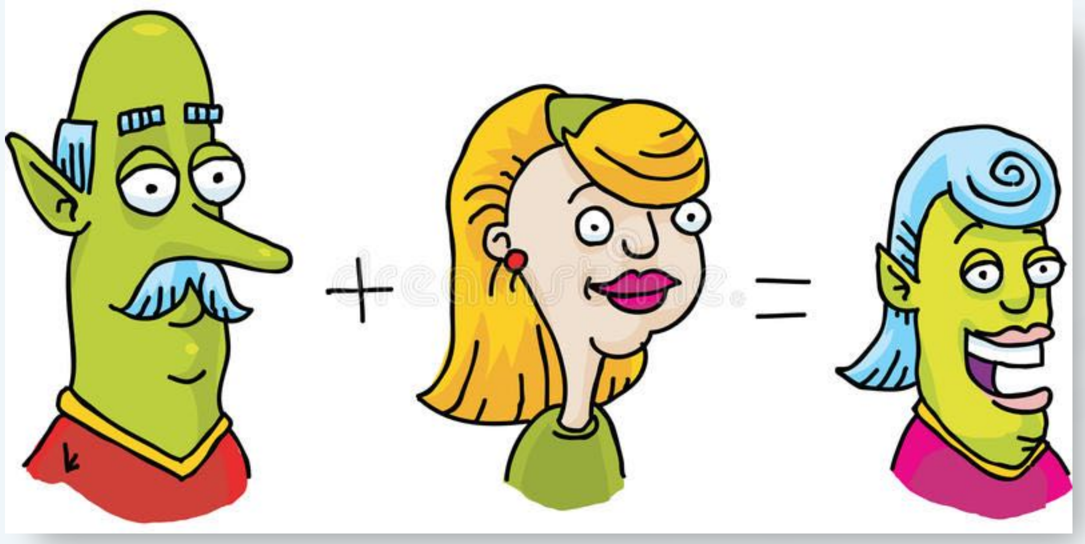
随机初始化

- 随机生成一系列的坐标值、颜色值和透明度
- 将这些值组合成一个个**随机**的三角形。
- 每128个三角形组成一个初始的个体
- 作为遗传算法最初的父代种群。



遗传和变异

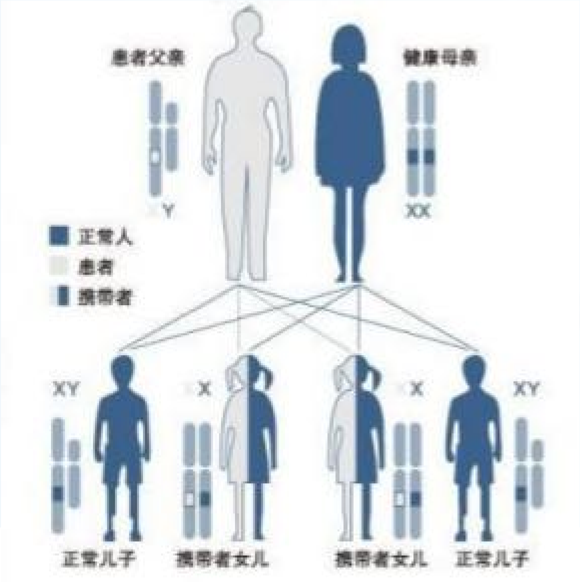
- 有了父代之后，我们需要通过父代繁衍获得新的子代
- 这个过程就包括了遗传和变异。



遗传和变异

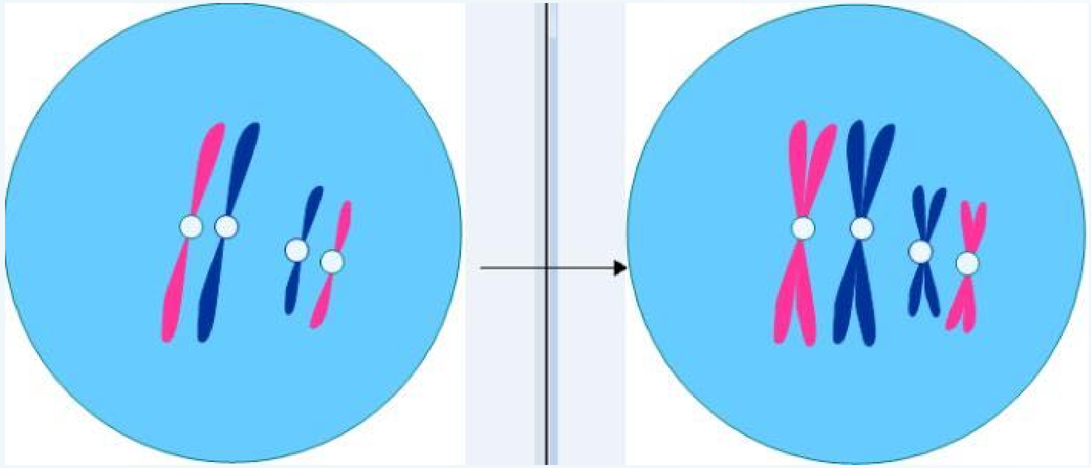
- 遗传
  - **复制**：染色体复制
  - **交叉**：染色体自由组合、交叉变换

没有产生新形状



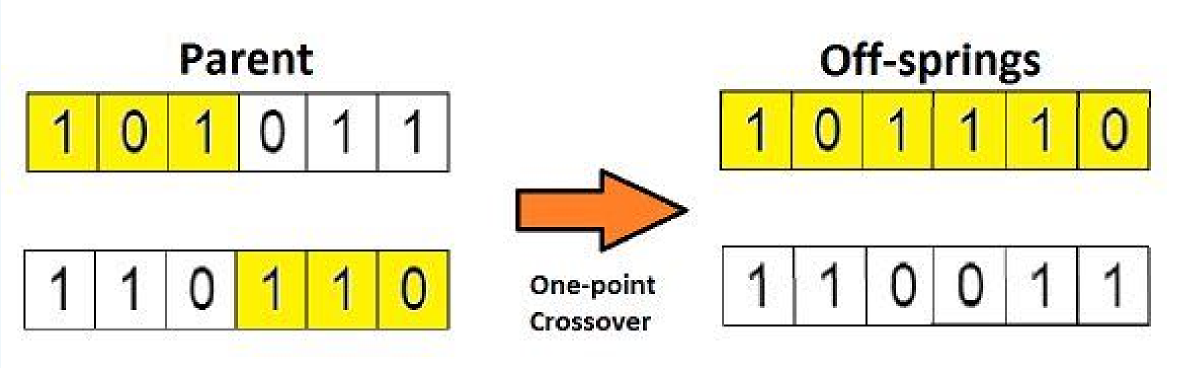
复制

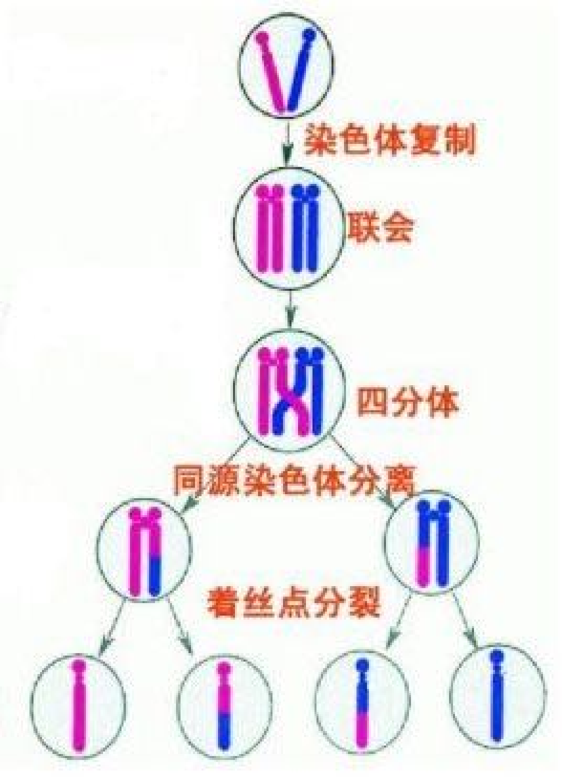
- **目的：保留优良染色体**
  （不保留的将被淘汰）
- **方式：将适应度高的染色体原封不动地复制给子代**



交叉

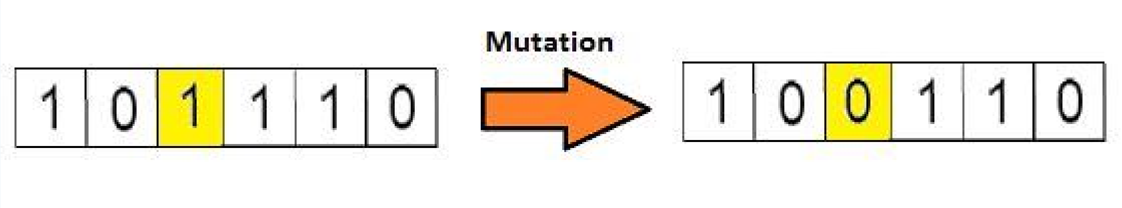
- 将来自父母双方的染色体组合在一起，得到新的染色体，但性状不变。

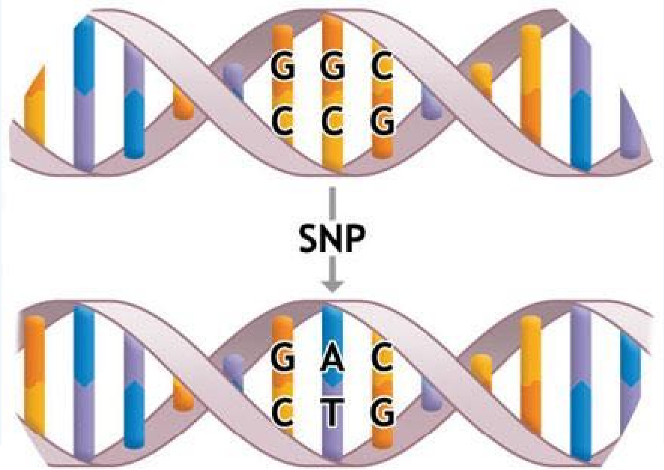


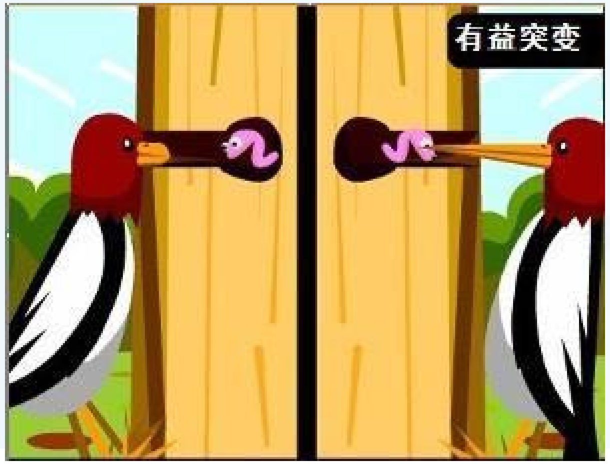


变异

- 由基因突变产生新的基因
- 已有的基因可能不是最优的基因

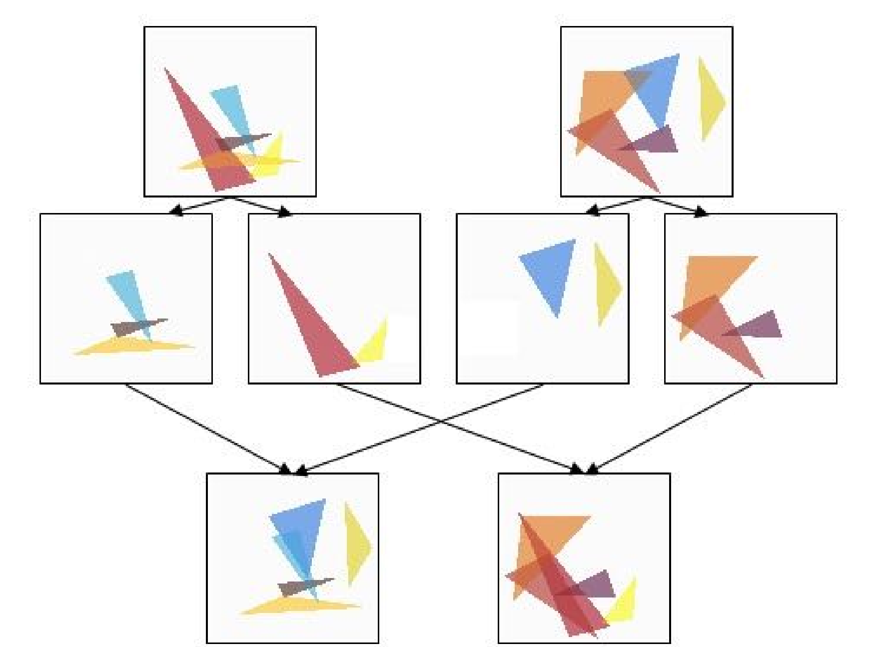






三角形的交叉

- 对于现有的父类
- 可以选择直接遗传继承
- 也可以选择交换某些三角形进行交叉

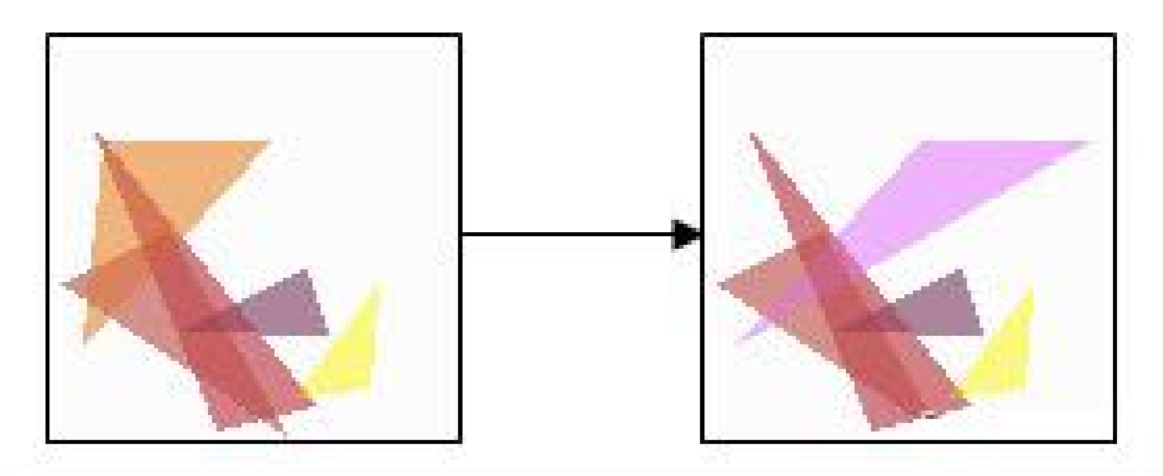


三角形的变异

- **将某个父类中的部分三角形进行变异**

  完全随机变异。

  或在原有基础上进行稍微改变。



控制变异的概率

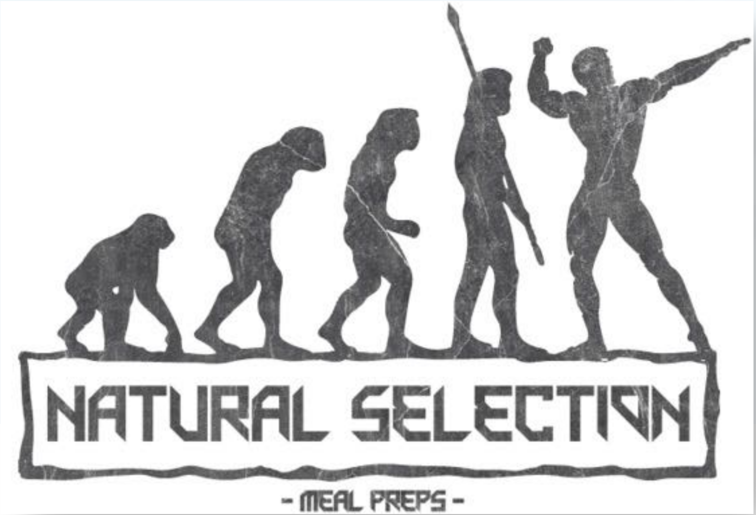
- 自然界生物变异的概率是不确定的。
- 对于三角形可以设置不同变异方式的概率，来获得更好的求解效果。

遗传和变异：总结

- 遗传是直接得到来自亲代的性状；
- 变异是随机改变来自亲代的性状。
- 亲代通过遗传和变异来生成新的子代。

自然选择

- 达尔文提出的关于生物进化机理的自然选择学说。
- 在不断变化的生活条件下
- 具有有利变异的个体能生存下来并繁殖后代
- 具有不利变异的个体则逐渐被淘汰。



自然选择

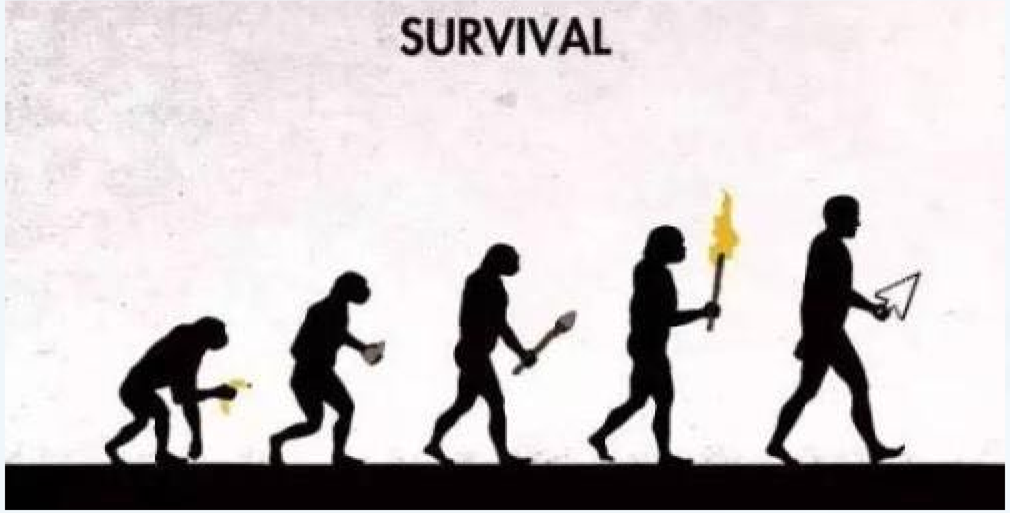
› **大自然靠什么选择优良物种呢？**

脖子长度

奔跑速度

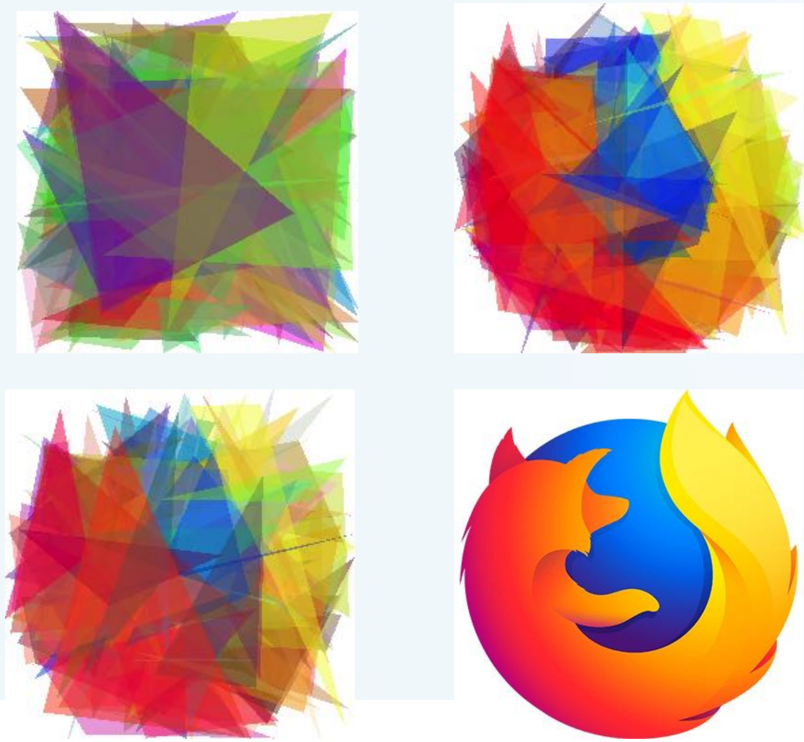
适应度函数

- 脖子长度、奔跑速度都是一个参数
- 我们可以利用**适应度函数**表示参数值的大小
- 判断个体是否应该被淘汰。



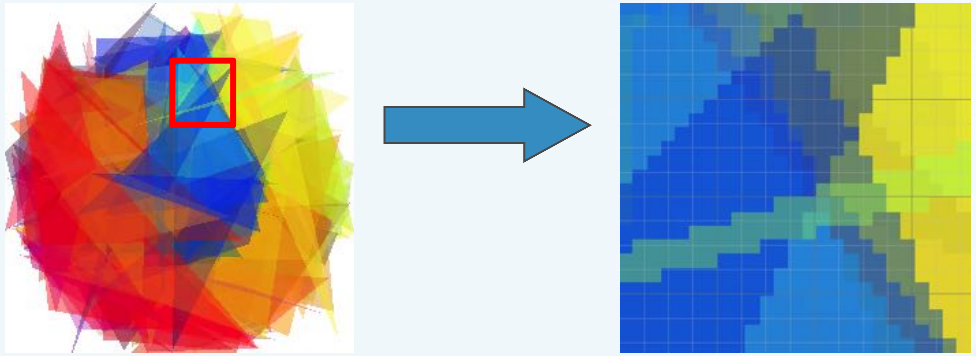
图片的适应度

- 要用三角形拼成的图片
- 该怎么计算适应度？



图片的适应度

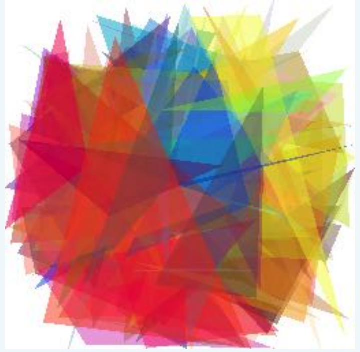
- 图片是由一系列像素点构成的
- 每个像素点有不同的R（红）、G（绿）、B（蓝）值来表示自己的颜色。



图片的适应度

- 对于每一个像素点，三个颜色通道分别计算当前图像A与目标图像B的差值。
- 将这个差值平方相加作为适应度函数

$$f = (R_A - R_B)^2 + (G_A - G_B)^2 + (B_A - B_B)^2$$



A

B

判定与淘汰

- 有了适应度函数，我们就能够代替大自然去决定哪一代保留，哪一代被淘汰。
- 如果**子代适应度 ＞ 亲代适应度**则子代取代亲代，反之子代被淘汰

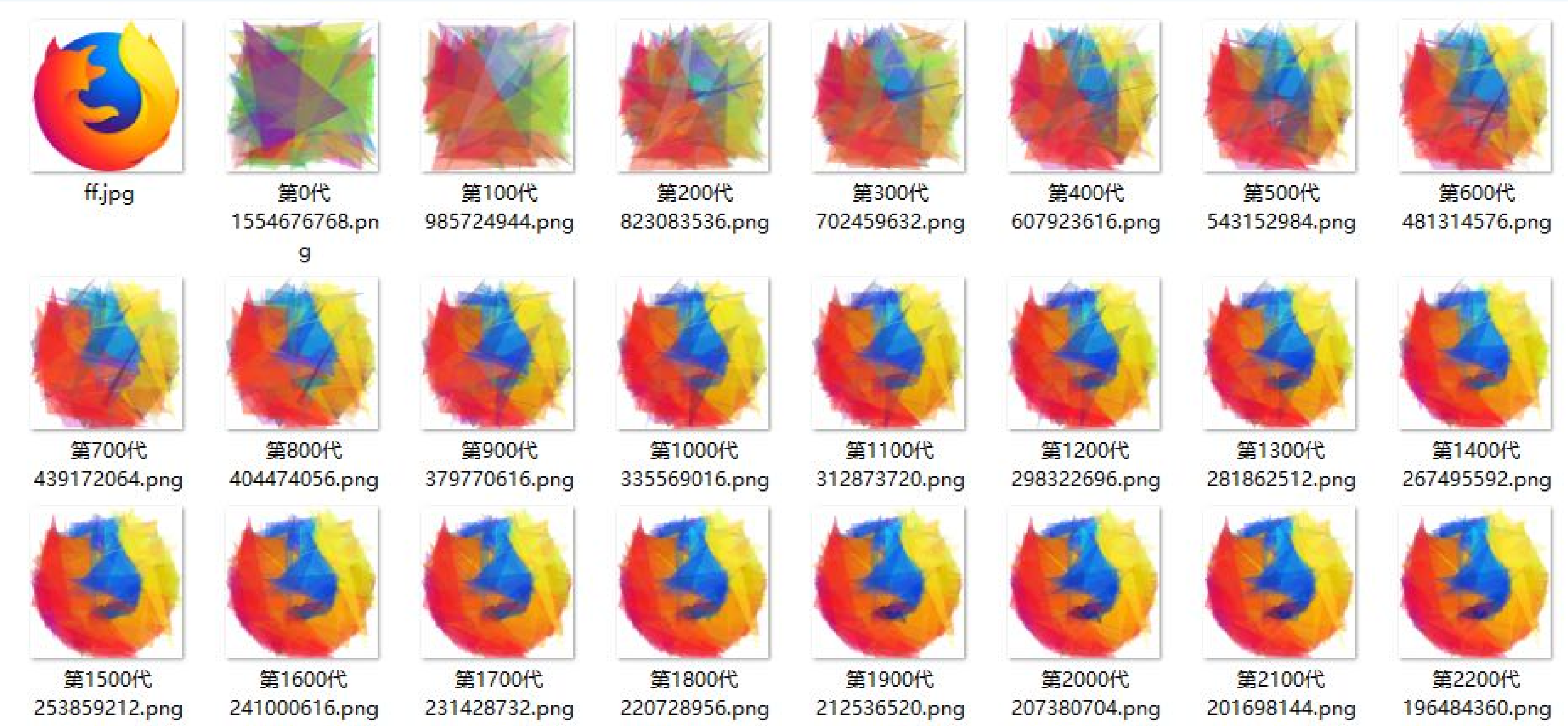
终止条件

- 当适应度函数**小于**一定阈值
- 或者
- 循环迭代的次数**超过**了一定的限制
- 遗传算法终止，所得到的最好后代即是我们要找的最优解

第29900代

总结

- 将当前图片与目标图片的差异量化为适应度函数。
- 模拟自然选择的过程，以适应度的高低来决定某一代结果的淘汰与生存。



看看代码