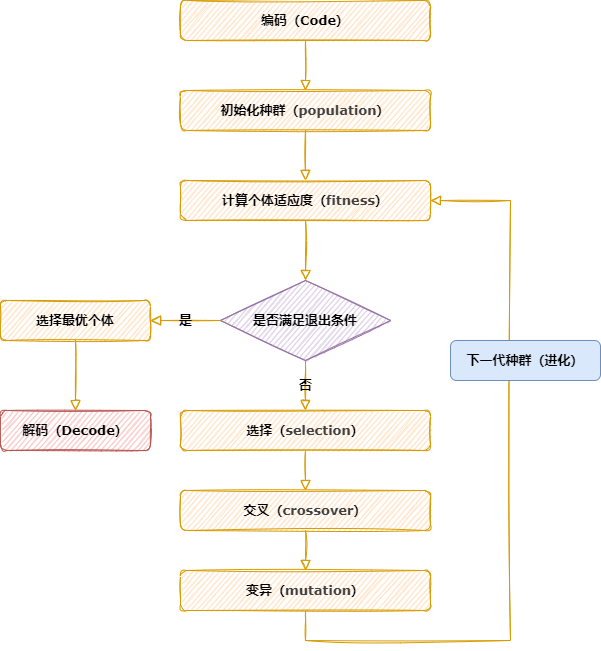
# 遗传算法学习资料



遗传算法核心的任务是要通过编码体系，给出解决方案的染色体表现规则，首先需要随机初始化一定数量的种群（population），而种群则由一定数目的个体(individual)构成。每个个体实际上是染色体(chromosome)，可以通过规则计算出适应度（fitness）。初代种群产生之后，按照优胜劣汰的进化原理，逐代进化产生出优秀的后代。

在每代进化过程中，根据个体的适应度大小来选择个体，并借助于自然遗传学的遗传算子（genetic operators）进行交叉（crossover）和变异（mutation），产生新的种群。末代种群中的最优个体经过解码（decoding），可以作为问题近似最优解。退出条件一般为达到最大的迭代次数，比如10000次，另外，就是适应度满足要去，比如达到0.99。基本的流程示意图如下所示：



实际的课程编排，由于涉及到大量的老师、班级、教室和课程等要素，因此非常的复杂，借助遗传算法也可能求不出最优解，而只是求出局部最优解，但是利用遗传算法辅助课程编排仍然是一个非常好的手段。一般来说，课程编排过程中，必须满足几个限制条件，否则，给出的课程安排是无效的，具体说明如下：

1. 同一时刻，一个教室只能开设一门课程；
2. 一个教室是有座位个数限制的，上课的学生总数不能超过教室座位数；
3. 同一个时刻，同一个老师或班级学生只能参与一门课程，而不能参与多个课程；
4. 教室分多媒体教室和普通教室，有的课程需要多媒体教室，因此，教室配置必须满足课程要求；

以上4条限制，都满足的情况下，给出的课程安排才是有效的，但请注意，不一定是最优的，它并未考虑优化条件，比如同一个老师，如果在一天按照多门课程，那么显然有点超负荷工作，或者同一门课程，在同一天，连续开设多次，这样对于老师和学生来说，都有点吃不消。