**运行结果：**

初始染色体组为: ['100100', '101010', '010101', '101011']

当前最大重量: 95

1: [[60, 35], [105, 60], [140, 75], [175, 95]]

2: [2, 1, 1]

新的初始染色体组为: ['100100', '101010', '010101']

染色体索引: 2

弹出后的染色体组: ['100100', '010101']

弹出的染色体: 101010

随机选择一个染色体: 100100

交叉点位置为: 3

新染色体组为: ['101100']

恢复到原染色体组为: ['100100', '101010', '010101']

染色体索引: 1

弹出后的染色体组: ['101010', '010101']

弹出的染色体: 100100

随机选择一个染色体: 101010

交叉点位置为: 4

新染色体组为: ['101100', '100110']

随机选择一个染色体: 101010

交叉点位置为: 3

新染色体组为: ['101100', '100110', '100010']

恢复到原染色体组为: ['100100', '101010', '010101']

染色体索引: 0

弹出后的染色体组: ['100100', '101010']

弹出的染色体: 010101

随机选择一个染色体: 100100

交叉点位置为: 1

新染色体组为: ['101100', '100110', '100010', '000100']

恢复到原染色体组为: ['100100', '010101', '101010']

下一代的染色体组为: ['101100', '100110', '100010', '000100']

迭代次数: 99

..................................

当前最大重量: 65

1: [[95, 55], [115, 65], [70, 40], [45, 25]]

2: [2, 0, 1, 1]

新的初始染色体组为: ['101100', '100110', '100010', '000100']

染色体索引: 3

弹出后的染色体组: ['101100', '100110', '000100']

弹出的染色体: 100010

随机选择一个染色体: 000100

交叉点位置为: 1

新染色体组为: ['100100']

恢复到原染色体组为: ['101100', '100110', '100010', '000100']

染色体索引: 2

弹出后的染色体组: ['101100', '100010', '000100']

弹出的染色体: 100110

恢复到原染色体组为: ['101100', '100110', '100010', '000100']

染色体索引: 1

弹出后的染色体组: ['100110', '100010', '000100']

弹出的染色体: 101100

随机选择一个染色体: 000100

交叉点位置为: 1

新染色体组为: ['100100', '100100']

随机选择一个染色体: 000100

交叉点位置为: 3

新染色体组为: ['100100', '100100', '101100']

恢复到原染色体组为: ['101100', '100110', '100010', '000100']

染色体索引: 0

弹出后的染色体组: ['101100', '100110', '100010']

弹出的染色体: 000100

随机选择一个染色体: 100010

交叉点位置为: 4

新染色体组为: ['100100', '100100', '101100', '000110']

恢复到原染色体组为: ['101100', '100110', '000100', '100010']

下一代的染色体组为: ['100100', '100100', '101100', '000110']

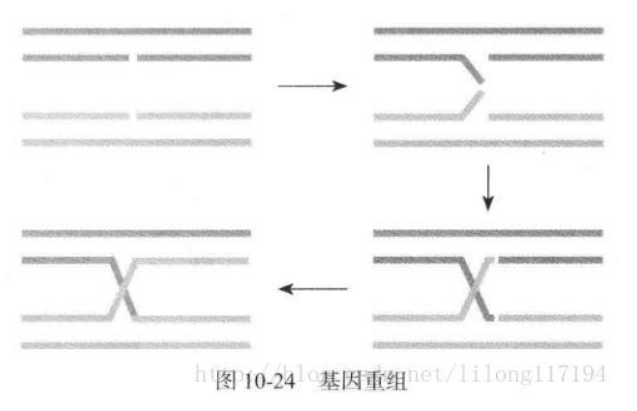
迭代次数: 98

..................................

当前最大重量: 55

各染色体总价值为: [[60, 35], [60, 35], [95, 55], [100, 55]]

['100100', '100100', '101100', '000110']

选用的是基因重组的方法，如下图：

整个过程是先计算适应度（总价值），如果满足给定的条件就继续进行，然后再筛选合适的染色体，生成新的初始染色体组，紧接着运行crossover函数进行基因重组生成下一代染色体组，在规定的迭代次数里进行迭代，然后每一代的适应度函数的值都比上一代的适应度更好，如果发现连续几代的适应度函数值基本不增加或者甚至减少的情况，那说明函数已经收敛了，在迭代完成后，最后得到的就是在规定迭代次数里最优的解了。

在这次运行中，在背包重量限制为80的情况下，有限的迭代次数中，最优秀的总价值为[100, 55],对应染色体为000110，物品组合为4、5

实际上，如果迭代次数足够多，最终的最优解应该是最大的适应度（总价值）为130的染色体组合，即物品2、3、6的组合。