

```
In [1]: import random

def cruzamento(ancestral1, ancestral2):
    tamanho = len(ancestral1)
    filho = []

    indice1 = 0
    indice2 = 0

    while(len(filho) != tamanho):
        moeda = (random.uniform(0, 1) < 0.5)
        if(moeda):
            if(indice1 < tamanho):
                convidado = ancestral1[indice1]
                indice1 += 1
                if(convidado not in filho):
                    filho.append(convidado)
            else:
                if(indice2 < tamanho):
                    convidado = ancestral2[indice2]
                    indice2 += 1
                    if(convidado not in filho):
                        filho.append(convidado)
        return filho

taxa_de_mutacao = 0.3

def mutacao(individuo):
    if(random.uniform(0, 1) > taxa_de_mutacao):
        return individuo
    indice1 = random.randint(0, len(individuo) - 1)
    indice2 = random.randint(0, len(individuo) - 1)
    aux = individuo[indice1]
    individuo[indice1] = individuo[indice2]
    individuo[indice2] = aux
    return individuo

total_convitados = 15

m_conflitos = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 1,
def conflitos(individuo):
    contagem = 0
    tamanho = len(individuo)
    for i in range(tamanho):
        indice = individuo[i]
        indice_proximo = individuo[(i + 1)%tamanho]
        indice_anterior = individuo[(i - 1)%tamanho]
        contagem = contagem + m_conflitos[indice][indice_proximo]
        contagem = contagem + m_conflitos[indice][indice_anterior]

    return contagem

tamanho_populacao = 200
```

```
def populacao_inicial():
    individuos = []

    for i in range(tamanho_populacao):
        individuo = []
        for j in range(total_convidados):
            individuo.append(j)

        for j in range(total_convidados):
            indice1 = random.randint(0, len(individuo) - 1)
            indice2 = random.randint(0, len(individuo) - 1)
            aux = individuo[indice1]
            individuo[indice1] = individuo[indice2]
            individuo[indice2] = aux

        individuos.append(individuo.copy())

    return individuos

def escolhe(populacao):
    tamanho = len(populacao)
    indice1 = random.randint(0, tamanho - 1)
    indice2 = random.randint(0, tamanho - 1)
    valor1 = conflitos(populacao[indice1])
    valor2 = conflitos(populacao[indice2])
    if(valor1 < valor2):
        return populacao[indice1]
    return populacao[indice2]

n_iteracoes = 100

populacao = populacao_inicial()

for i in range(n_iteracoes):
    proxima_geracao = []
    for j in range(tamanho_populacao):
        x = escolhe(populacao)
        y = escolhe(populacao)
        filho = cruzamento(x, y)
        filho = mutacao(filho)
        proxima_geracao.append(filho)
    populacao = proxima_geracao

for i in populacao:
    print(conflitos(i))
```

7
5
4
9
4
4
6
6