

1-

A) Critério Guloso: O critério guloso a ser utilizado para inserir os vértices na solução, a partir do segundo, é o da aresta de menor peso. Ou seja, a cada passo, o algoritmo escolhe o vértice mais próximo (ou seja, a aresta de menor peso) que ainda não foi visitado.

B) Construção da Solução: Suponha que escolhemos o vértice 'A' como nosso ponto de partida. O vértice mais próximo é 'B', então vamos para 'B'. Em seguida, do vértice 'B', vamos para o vértice 'C', que é o próximo mais próximo. Repetimos esse processo até visitar todos os vértices. Finalmente, retornamos ao vértice 'A' para completar o ciclo.

C) Custo da Solução: O custo da solução é a soma dos pesos de todas as arestas no ciclo formado. Isso pode ser calculado somando-se os pesos conforme avançamos de um vértice para outro no ciclo.

2-

A) Estratégia de Busca Local: A estratégia usada pela busca local é a de troca de dois vértices. Em cada passo, o algoritmo escolhe dois vértices e troca suas posições no ciclo. A ideia é que, ao fazer isso, podemos potencialmente diminuir o custo total do ciclo.

B) Execução da Busca Local: Suponha que começamos com a solução da questão 1. Para o primeiro passo, escolhemos dois vértices aleatoriamente e trocamos suas posições. Por exemplo, podemos trocar 'B' e 'C'. No segundo passo, fazemos outra troca. Por exemplo, podemos trocar 'D' e 'E'. Em cada passo, verificamos se a nova solução tem um custo menor do que a solução atual. Se tiver, atualizamos nossa solução.

C) Custo das Soluções: O custo das soluções encontradas é a soma dos pesos de todas as arestas no ciclo formado, assim como na questão 1. Isso pode ser

calculado somando-se os pesos conforme avançamos de um vértice para outro no ciclo.

4-

1. População Inicial (Passo A): 4 soluções iniciais aleatórias. Aqui, cada solução é uma sequência binária representando quais itens estão na mochila (1 para incluído, 0 para excluído).

- Solução 1: 10011001001010110001 (Valor: 356, Peso: 475)
- Solução 2: 11000001110000010000 (Valor: 269, Peso: 339)
- Solução 3: 00110110101010001010 (Valor: 321, Peso: 358)
- Solução 4: 00001010000010001010 (Valor: 104, Peso: 149)

2. Seleção de Pares de Pais (Passo B): Escolher aleatoriamente dois pares de soluções pais da população atual.

- Par 1: Solução 1 e Solução 3
- Par 2: Solução 2 e Solução 4

3. Crossover (Passo C): Executar o operador de crossover nos pares de pais. Vou usar um ponto de corte no meio dos cromossomos:

- Filho 1 do Par 1: 10011010101010010001
- Filho 2 do Par 1: 00110100101010110010
- Filho 1 do Par 2: 11001010000010001010
- Filho 2 do Par 2: 00000001110000010000

4. Mutação (Passo D): Escolher aleatoriamente um dos filhos gerados no passo anterior e realizar uma mutação de flip (inversão de um bit):

- Vamos escolher o Filho 2 do Par 1: 00110100101010110010
- Após a mutação (inversão de um bit aleatório): 00110100101010110011

5. Verificação de Restrição de Peso (Passo E): Verificar se as soluções filhas quebram a restrição de peso máximo. No caso, a solução do Filho 2 do Par 1 não quebra a restrição de peso, então não é necessário remover itens.

6. Cálculo da Qualidade (Passo F): Calcular o valor total das soluções filhas:

- Valor do Filho 1 do Par 1: 280
- Valor do Filho 2 do Par 1 (após mutação): 281
- Valor do Filho 1 do Par 2: 234
- Valor do Filho 2 do Par 2: 104

7. Seleção Natural (Passo G): Selecionar as 4 melhores soluções para formar a próxima população. Neste caso, selecione as duas soluções filhas e as duas soluções originais com base no valor.

- População da Próxima Geração:
- Solução 1: 10011001001010110001 (Valor: 356, Peso: 475)
- Solução 2 (Filho 2 do Par 1): 00110100101010110011 (Valor: 281, Peso: 355)
- Solução 3: 00110110101010001010 (Valor: 321, Peso: 358)
- Solução 4: 11001010000010001010 (Valor: 234, Peso: 299)

8. Passo H:

Nossa população atual é a seguinte:

- Solução 1: 10011001001010110001 (Valor: 356, Peso: 475)
- Solução 2 (Filho 2 do Par 1): 00110100101010110011 (Valor: 281, Peso: 355)
- Solução 3: 00110110101010001010 (Valor: 321, Peso: 358)
- Solução 4: 11001010000010001010 (Valor: 234, Peso: 299)

Vamos repetir os passos B a G para gerar a próxima geração:

1. Seleção de Pares de Pais (Passo B): Escolher aleatoriamente dois pares de soluções pais da população atual:

- Par 1: Solução 1 e Solução 2
- Par 2: Solução 3 e Solução 4

2. Crossover (Passo C): Executar o operador de crossover nos pares de pais:

- Filho 1 do Par 1: 10011000101010110011
- Filho 2 do Par 1: 00110101001010001001
- Filho 1 do Par 2: 00101010001010001010
- Filho 2 do Par 2: 11010110101010011010

3. Mutação (Passo D): Escolher aleatoriamente um dos filhos gerados no passo anterior e realize uma mutação de flip (inversão de um bit):

- Vou escolher o Filho 2 do Par 2: 11010110101010011010
- Após a mutação (inversão de um bit aleatório): 11010110101010011011

4. Verificação de Restrição de Peso (Passo E): Verificar se as soluções filhas quebram a restrição de peso máximo. No caso, a solução do Filho 2 do Par 2 não quebra a restrição de peso, então não é necessário remover itens.

5. Cálculo da Qualidade (Passo F): Calcule o valor total das soluções filhas:

- Valor do Filho 1 do Par 1: 330
- Valor do Filho 2 do Par 1: 267
- Valor do Filho 1 do Par 2: 273
- Valor do Filho 2 do Par 2 (após mutação): 269

6. Seleção Natural (Passo G): Selecionar as 4 melhores soluções para formar a próxima população com base no valor:

- População da Próxima Geração:
- Solução 1: 10011001001010110001 (Valor: 356, Peso: 475)
- Solução 3: 00110110101010001010 (Valor: 321, Peso: 358)
- Filho 1 do Par 1: 10011000101010110011 (Valor: 330, Peso: 425)
- Filho 2 do Par 2 (após mutação): 11010110101010011011 (Valor: 269, Peso: 354)

Repetindo o Passo B ao H novamente, temos a população de terceira geração.

Nossa população atual é a seguinte:

Solução 1: 10011001001010110001 (Valor: 356, Peso: 475)

Solução 3: 00110110101010001010 (Valor: 321, Peso: 358)

Filho 1 do Par 1: 10011000101010110011 (Valor: 330, Peso: 425)

Filho 2 do Par 2 (após mutação): 11010110101010011011 (Valor: 269, Peso: 354)

Vamos repetir os passos B a G para gerar a próxima geração:

Seleção de Pares de Pais (Passo B): Escolha aleatoriamente dois pares de soluções pais da população atual:

Par 1: Solução 1 e Solução 3

Par 2: Filho 1 do Par 1 e Filho 2 do Par 2 (após mutação)

Crossover (Passo C): Execute o operador de crossover nos pares de pais:

Filho 1 do Par 1: 10011010101010001010

Filho 2 do Par 1: 00110101001010110001

Filho 1 do Par 2: 10011010101010011011

Filho 2 do Par 2 (após mutação): 11010100101010011011

Mutação (Passo D): Escolha aleatoriamente um dos filhos gerados no passo anterior e realize uma mutação de flip (inversão de um bit):

Vamos escolher o Filho 2 do Par 1: 00110101001010110001

Após a mutação (inversão de um bit aleatório): 00110101001010110011

Verificação de Restrição de Peso (Passo E): Verifique se as soluções filhas quebram a restrição de peso máximo. No caso, a solução do Filho 1 do Par 2 não quebra a restrição de peso, então não é necessário remover itens.

Cálculo da Qualidade (Passo F): Calcule o valor total das soluções filhas:

Valor do Filho 1 do Par 1: 336

Valor do Filho 2 do Par 1 (após mutação): 268

Valor do Filho 1 do Par 2: 324

Valor do Filho 2 do Par 2 (após mutação): 324

Seleção Natural (Passo G): Selecione as 4 melhores soluções para formar a próxima população com base no valor:

População da Próxima Geração:

Solução 1: 10011001001010110001 (Valor: 356, Peso: 475)

Filho 1 do Par 1: 10011010101010001010 (Valor: 336, Peso: 425)

Filho 1 do Par 2: 10011010101010011011 (Valor: 324, Peso: 419)

Filho 2 do Par 2 (após mutação): 11010100101010011011 (Valor: 324, Peso: 354)

Neste ponto, temos a população da terceira geração. Podemos continuar repetindo os passos B a G para gerar gerações subsequentes até atingir um critério de parada, como um número máximo de gerações, convergência para uma solução ótima ou outro critério de sua escolha. A cada geração, as soluções devem melhorar em termos de valor, pois o algoritmo genético busca otimizar o valor da mochila respeitando a restrição de peso máximo.