

Universidade Federal do Ceará
Campus de Quixadá
QXD0153 - Desafios de Programação

Lista 4 - Programação Dinâmica

1. Dado n amigos, cada um amigo pode ficar sozinho ou fazer dupla uma única vez com algum outro amigo. Encontre o número total de maneiras em que os amigos podem ser organizados.

Entrada: $n = 3$

Saída : 4

Explicação:

$\{1\}, \{2\}, \{3\}$: 3 amigos sozinhos $\{1\}, \{2,3\}$: 1 sozinho, 1 dupla com 2 e 3

$\{1,2\}, \{3\}$: 1 dupla com 1 e 2, 3 sozinho $\{1,3\}, \{2\}$: 1 dupla com 1 e 3, 2 sozinho

2. Dado um madeira de n metros e um vetor de preços que contém os preços de todas as peças de tamanho menor que n . Determine o valor máximo obtido cortando a madeira e vendendo as peças. Por exemplo, se o comprimento da madeira é 8 e o vetor de preços abaixo então o valor máximo é 22 (cortando em dois pedaços de comprimento 2 e 6).

comprimento	1	2	3	4	5	6	7	8
preço	1	5	8	9	10	17	17	20

3. Pedro Henrique trabalha no caixa de um supermercado. Ele precisa passar o troco milhares de vezes durante o dia. Só que ele apaixonado por moedas e quer entregar o menor números de moedas possível a cada cliente. Dado um valor N e um estoque infinito de cada uma das M moedas de diferentes valores m_1, m_2, \dots, m_M , desenvolva um algoritmo que encontra quais e quantas moedas você deve entregar ao cliente de modo que o total de moedas seja o mínimo possível. Por exemplo, dado $N=49$ e 5 moedas de valores 1,5,6,13,27 o menor número de moedas para $N=49$ são duas moedas de valor 5, duas moedas de 6 e um moeda de 27.
4. Enoque e Leonardo encontraram um tesouro secreto no campus de Quixadá. O tesouro é composto por n itens de valores diferentes v_1, v_2, \dots, v_n . Eles querem saber se é possível particionar os itens do tesouro em dois subconjuntos de tal maneira que ambos recebam o mesmo valor. Por exemplo, se eles encontrarem um tesouro composto por 3 itens de valores 1, 5, 11 e 5. Esse tesouro pode ser particionado como $\{1, 5, 5\}$ e $\{11\}$
5. Pedro Olímpio tem uma capacidade impressionante, dada uma sequência qualquer, ele consegue dizer o tamanho da maior subsequência palindrome contida na sequência dada. Desenvolva um algoritmo com a mesma capacidade do Pedro Olímpio. Por exemplo, se a sequência é "BBABCBCAB", então o tamanho da maior subsequência palindrome é 7 e "BABCBAB" é um subsequência palindrome com esse tamanho.
6. Nas suas horas vagas, Décio gosta de descobrir o número mínimo de caracteres que precisam ser inserido em um string para que ela torne-se palindrome. Porém, ele faltou a aula de Programação Dinâmica e agora não sabe resolver este problema de maneira ótima. Desenvolva um algoritmo para o Décio que dada uma string qualquer, devolva o número mínimo de caracteres que precisam ser inseridos. Por exemplo, dada a string $s = \text{"Ab3bd"}$, o número mínimo de caracteres que precisa ser inserido é 2 e podemos obter "dAb3bAd" ou "Adb3bdA".

7. Dado um vetor de n inteiros positivos. Desenvolva um algoritmo para encontrar a subsequência de soma máxima tal que os inteiros na subsequência estão ordenados em ordem crescente. Por exemplo, se a entrada é $\{1,101,2,3,100,4,5\}$, então a saída deve ser $\{1,2,3,100\}$.
8. Daiane propôs um novo jogo para Décio chamado particionamento palíndromo. Dada uma string, o particionamento palíndromo é um particionamento da string que toda substring da partição é um palíndromo. Por exemplo, "aba|b|bbabb|a|b|aba" é um particionamento palíndromo de "ababbbabbababa". Desenvolva um algoritmo que encontra o número mínimo de partições necessárias de uma partição palíndromo.
9. Um número é não-decrescente se todo dígito (exceto o primeiro) é menor ou igual ao dígito anterior. Por exemplo, 223, 4455567 e 899 são números não-decrescentes. Dado o número de dígitos n , devolva a quantidade de números não-decrescentes de n dígitos.
10. Dada uma string s , a tarefa é contar o número de subsequências da forma $a^i b^j c^k$, onde $i \geq 1$, $j \geq 1$ e $k \geq 1$. Duas subsequências são consideradas diferentes se os índices escolhidos da sequência inicial para formar as duas subsequências são diferentes.

Entrada : abcabc

Saída : 7

Explicação: As subsequências são abc, abc, abbc, aabc, abcc, abc e abc

11. Existem 100 diferentes tipos de bonés cada um tem um identificador único de 1 a 100. Além disso, existem n pessoas cada um com uma coleção de bonés. Um dia todas essas pessoas decidem ir a uma festa usando boné, porém elas decidiram que nenhuma delas vai usar um boné do mesmo tipo. Desenvolva um algoritmo que encontre o número total de formas tais que nenhuma delas está usando um boné do mesmo tipo. A primeira linha da entrada contém o valor n , as n linhas seguintes contém as coleções de cada uma das pessoas.

Entrada

3

5 100 1 // Coleção da primeira pessoa

2 // Coleção da segunda pessoa

5 100 // Coleção da terceira pessoa

Saída

4

Explicação: Todas as maneiras possíveis são (5,2,100), (100,2,5), (1,2,5) e (1,2,100).

Descreva a estrutura recursiva do seguinte problema:

$Count_{S,j}$ = Número de maneiras que podemos atribuir os bonés identificados identificados $[1..j]$ para as pessoas do conjunto S .

Considere os seguintes casos na estrutura recursiva:

- (a) O boné j não faz parte da coleção de nenhuma pessoa do conjunto S .
- (b) O boné j faz parte da coleção de um ou mais pessoas do conjunto S .

Defina os seguintes casos bases: $Count_{\emptyset,j}=?$ e $Count_{S,0}=?$