

MC458A - Laboratório 5 – Problema do troco

1 Introdução

Sempre que viaja para outros países, o Professor Sabin Ada prefere levar moedas e notas do país destino do que usar cartões de crédito. Não se sabe exatamente a razão para isso: alguns de seus amigos acham que ele é ligeiramente (ou muito) paranoico e não confia em bancos, outros acham é apenas uma mania atrelada a seu espírito aventureiro. De qualquer forma, o Professor Sabin Ada gosta de comprar lembrancinhas em suas viagens, para ele ou para amigos e parentes. É comum ele se ver na seguinte situação ao tentar comprar algum item.

Vendedor: São 14 dólares!

Professor Sabin Ada: Aqui está uma nota de 20.

Vendedor: Humm... Lamento, estou sem troco.

Professor Sabin Ada: Tudo bem! Aqui tem uma nota de 10 e uma de 5. Pode ficar com o troco!

Em outras palavras, o Professor Sabin Ada quer comprar um item de um certo valor. No entanto, os tipos de nota que ele tem podem não permitir a ele que pague o valor exato do item. Ele gostaria então de minimizar a quantia que ele pagaria (claro, cobrindo o preço do item). Além disso, ainda pagando a quantia mínima, Professor Sabin Ada quer também minimizar o número de notas usadas que ele usaria para pagar pelo item.

Você deve ajudar o Professor Sabin Ada projetando um algoritmo de programação dinâmica para resolver seu problema!

Observação: suponha que o Professor Sabin Ada tem um número suficientemente grande de notas de cada tipo. Ou seja, ele nunca fica sem notas de um determinado valor, para efeito da resolução do problema.

Dica!?

Considere o seguinte problema ligeiramente diferente. Suponha que o item tem valor U e que o Professor Sabin Ada quer saber se é possível pagar o **valor exato** U com as moedas que tem e usando o **menor número possível de moedas**. Seja $r(n, U)$ o menor número das n moedas que devem ser usadas para pagar exatamente o valor U . Definimos $r(n, U) = +\infty$ se não é possível pagar exatamente. Por exemplo, se $U = 6$, $n = 4$ e as notas são 1, 2, 3, 4, então há várias formas de pagar: por exemplo, $6 = 1 + 2 + 3$ ou $6 = 2 + 4$ ou $6 = 1 + 1 + 1 + 3$. Não há como pagar com uma única nota; assim, a segunda maneira usa

o menor número de notas (duas). Logo, $r(4, 6) = 2$ nesta instância. Se $U = 17$, $n = 4$ e as moedas são 6, 7, 8, 20, então $r(4, 17) = +\infty$.

Tente descobrir uma recorrência para $r(n, U)$: se o valor da n -ésima nota for maior que U , então ela não pode ser usada; caso contrário, a solução ótima ou usa a nota ou não (obviamente). Você então terá dois subproblemas do mesmo tipo. A recorrência que você deveria obter é semelhante à recorrência do Problema da Mochila (que vimos/veremos) em aula. Entretanto, note que aqui a mesma nota pode ser usada várias vezes.

Voltando ao problema do Professor Sabin Ada, suponha que o valor do item seja V . Para resolver o problema, basta descobrir o menor U tal que $U \geq V$ e $r(n, U) < +\infty$. Por exemplo, se $V = 17$ e $n = 4$ e as moedas são 6, 7, 8, 20, então $U = 18$ e $r(n, 18) = 3$. De fato, não é possível pagar exatamente 17 com essas notas; além disso, $18 = 6 + 6 + 6$ e não é possível pagar 18 com menos notas.

Observações:

- Você não precisa fazer exatamente desta maneira, mas não me parece que seja possível resolver o problema sem usar algum tipo de programação dinâmica.
- Suponha que o Professor Sabin Ada tem um número suficientemente grande de notas de cada tipo.
- **Não** serão aceitas soluções com recursão e memorização. Quem fizer assim, receberá nota ZERO no trabalho.
- Os valores das notas que o Professor Sabin Ada possui não são necessariamente valores de notas que pertencem ao sistema monetário brasileiro. As notas podem ter qualquer valor dentro do intervalo especificado na próxima seção.

2 Especificação de Entrada e Saída

Entrada

A primeira linha contém um inteiro V ($1 \leq V \leq 50.000$) especificando o valor do item. A segunda linha contém um inteiro n ($1 \leq n \leq 100$) especificando o número de tipos de notas que ele tem. A terceira linha contém n inteiros positivos, cada um representando o valor v_i de cada nota ($1 \leq v_i \leq 2000$).

Saída

Imprima, em uma única linha, a quantia mínima que deve ser paga e o número mínimo de notas usadas. Os valores devem estar separados por um espaço.

Exemplos

Entrada	Saída
13	13 3
3	
3 5 7	

Entrada	Saída
20	25 1
4	
50 30 25 15	

3 Implementação e Submissão

- A solução deverá ser implementada em C, C++, Pascal ou Python 2/Python 3. Só é permitido o uso de bibliotecas padrão. Não é permitido o uso de nenhuma flag/diretiva de otimização.
- O programa deve ser submetido no SuSy, com o nome principal **t5** (por exemplo, t5.c).
- O número máximo de submissões é 20.
- A tarefa contém 10 testes abertos e 10 testes fechados. A nota será proporcional ao número de acertos nos testes fechados.

A solução pode ser submetida até o dia 18/11/21.