

Aluno: André Santos Rocha

Ra: 235887

Problema 1: A - Maximum Subarray Sum

O problema nos fornece um array de n inteiros, devemos devolver a maior soma de um subarray contido no array fornecido.

1.1 Ideia de solução

Para este problema, utilizamos o algoritmo de Kadane. Esse algoritmo consiste nos seguintes passos: inicializamos duas variáveis `soma_atual = 0` e `soma_max = INT_MIN`; percorremos cada elemento `v[i]` do array de entrada fazendo `soma_atual += v[i]`; atualizamos o nosso `soma_max` para ser o máximo entre `soma_atual` e `soma_max`; se `soma_atual < 0`, atualizamos `soma_atual` para 0. Esse algoritmo é válido. Se estivermos computando uma soma positiva, seguimos atualizando `soma_max`, enquanto essa soma aumentar. Caso estivermos computando um valor de soma negativa, atualizamos `soma_atual` para zero (o que é válido, pois se o próximo elemento a ser somado for positivo ou negativo, é melhor somá-lo a 0 do que a um número negativo). Por fim devolvemos o valor armazenado em `soma_max`.

1.2 Detalhes de implementação

A inicialização de `soma_max = INT_MIN` é válida porque nosso problema só contém números inteiros na entrada. Em outros casos, seria preciso adotar outras constantes.

Problema 2: G - Money Sums

O problema fornece n valores inteiros positivos. Devemos devolver todas as possíveis somas distintas possíveis de serem produzidas com os valores fornecidos.

2.1 Ideia de solução

Para este problema, criaremos um set somas para nos auxiliar. Para cada valor inteiro i fornecido na entrada, percorremos o set, somando esse inteiro a todos os valores de soma já contidos no set e acrescentamos cada uma dessas somas no set. Por fim, adicionamos o próprio i no set. Essa solução é válida. Ao inserir um elemento no set, sabemos que todos os outros elementos serão somados a ele e essas somas também serão inseridas no set, o que nos garante todas as somas possíveis de dois elementos. Agora, perceba também que, como essas somas foram inseridas no set, todos os elementos também serão somados a elas, o que nos garante todas as somas possíveis de 3 elementos. E podemos prolongar esse mesmo raciocínio até a maior soma possível dos n elementos, o que nos mostra que a

solução é válida. Ao final do programa basta devolver o tamanho do set e todos os valores nele contidos.

2.2 Detalhes de implementação

A utilização do set se deve ao fato dessa estrutura não permitir elementos repetidos, dessa forma garantimos que todas as somas sejam distintas.