

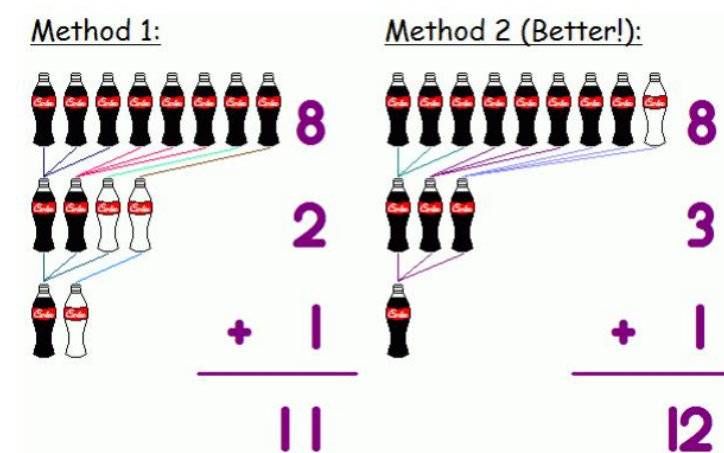
Você vê a seguinte oferta especial pela loja de conveniência:

"Uma garrafa de Choco Cola para cada 3 garrafas vazias retornadas"

Agora você decide comprar algumas (digamos N) garrafas de refrigerante da loja. Você gostaria de saber como você pode obter o máximo de choco cola deles.

A figura abaixo mostra o caso onde $N = 8$. O método 1 é o modo padrão: depois de terminar o seu 8 garrafas de cola, você tem 8 garrafas vazias. Tome 6 deles e você recebe 2 novas garrafas de cola. Agora depois bebendo-os você tem 4 garrafas vazias, então você pega 3 delas para obter outra nova cola. Finalmente, você tem apenas 2 garrafas na mão, então você não pode mais usar cola nova. Portanto, você bebeu $8 + 2 + 1 = 11$ garrafas de cola.

Você pode realmente fazer melhor! No Método 2, você primeiro toma emprestado uma garrafa vazia do seu amigo, então você pode beber de $8 + 3 + 1 = 12$ garrafas de cola! Claro, você terá que Devolva sua garrafa vazia restante para o seu amigo.



Defina uma função `chococola :: Int -> Int` tal que `(cola n)` devolve o número de garrafas de cola bebidas usando o método 2.

`chococola 81 == 121`

`chococola 40 == 60`

`chococola 20 == 30`

`chococola 60 == 90`

Dica: Faça uma função auxiliar `garrafa :: Int -> Int -> Int` tal que (garrafas cheias vazias)

devolve o número de chocolates bebidas considerando que cheias representa o número de garrafas cheias e vazias representa o número de garrafas vazias.