

DESAFIO: Dragão de Agualoo

Forma de entrega: [link do programa salvo no Gist do Github](#)

Linguagens aceitas: Javascript, Java, C#, Python

Certo dia o Reino de Agualoo foi atacado por um dragão cuspidor de fogo de várias cabeças, ameaçando queimar o reino até as cinzas. Preocupado, o rei chamou seus cavaleiros para derrotar o dragão e salvar o reino.

Os cavaleiros explicaram: “Para derrotar o dragão, temos que cortar todas as suas cabeças. Cada cavaleiro consegue cortar só uma das cabeças do dragão. A ordem dos cavaleiros requer que, por cortar uma cabeça, um cavaleiro seja pago uma recompensa igual a uma moeda de ouro para cada centímetro da altura do cavaleiro.

Haverá cavaleiros o suficiente para derrotar o dragão? O rei chamou seus conselheiros para ajudar a decidir quantos e quais cavaleiros contratar, minimizando o custo de derrotar o dragão. Como um de seus conselheiros, você deve ajudar o rei. Isto é muito sério: se você falhar, o reino inteiro será reduzido a cinzas!

Entrada

Obs: o formato aqui está diferente do vídeo explicativo, mas são os mesmos dados. O importante é fazer a função que receba os dados e produza a saída desejada. Veja as assinaturas das funções ao final deste documento.

- n: o número de cabeças do dragão.
- m: o número de cavaleiros no reino.
- diameters: uma lista de inteiros que representam o diâmetro das cabeças do dragão, em centímetros.
- heights: uma lista de inteiros que especifica a altura dos cavaleiros de Agualoo, também em centímetros.

Saída

Imprima uma linha contendo o número mínimo de moedas de ouro que o rei precisará pagar para derrotar o dragão. Se não for possível que os cavaleiros de Agualoo derrotem o dragão, imprima a linha “Agualoo esta condenada!”.

Entrada 1	Saída 1
<pre>{ "n": 2, "m": 3, "diameters": [5, 4], "heights": [7, 8, 4] }</pre>	11

Entrada 2	Saída 2
<pre>{ "n": 2, "m": 1, "diameters": [5, 5], "heights": [10] }</pre>	Agualoo esta condenada!

Entrada 3	Saída 3
<pre>{ "n": 2, "m": 4, "diameters": [7, 2], "heights": [4, 3, 1, 2] }</pre>	Agualoo esta condenada!

Entrada 4	Saída 4
<pre>{ "n": 2, "m": 4, "diameters": [7, 2], "heights": [2, 1, 8, 5] }</pre>	10

Entrada 5	Saída 5
<pre>{ "n": 2, "m": 10, "diameters": [1234567, 2345], "heights": [12345610, 1, 123, 23564, 123456, 123, 2, 3, 2, 1] }</pre>	12369174

Assinaturas:

Javascript:

```
function agualoo(n, m, diameters, heights)
```

Java:

```
static void agualoo(int n, int m, int[] diameters, int[] heights)
```

C#:

```
static void Agualoo(int n, int m, int[] diameters, int[] heights)
```

Python:

```
def agualoo(n, m, diameters, heights)
```