

# Olimpíada Brasileira de Informática 2017

Nível Sênior – Fase 2: *Upsolving*

---

Prof. Edson Alves - UnB/FGA

2020

## 1. Cortando o Papel

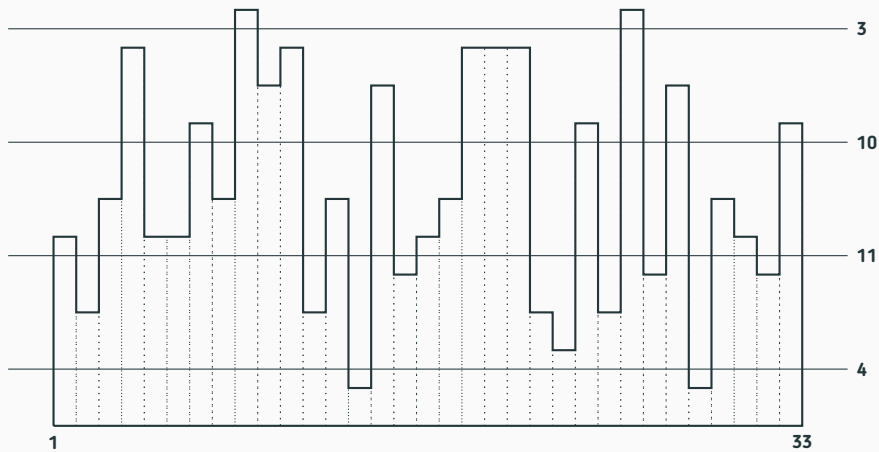
## **Cortando o Papel**

---

# Problema

Uma folha de papel é composta de uma sequência de retângulos com diferentes alturas mas com larguras fixas, tal que as bases dos retângulos estão assentadas em uma linha horizontal. A figura ilustra uma folha exemplo com 33 retângulos. Nós gostaríamos de fazer um único corte horizontal, com a ajuda de um estilete e uma régua, que maximize o número resultante de pedaços separados pelo corte. A figura mostra quatro diferentes cortes que resultariam, respectivamente, em 4, 11, 10 e 3 pedaços.

# Problema



## Entrada

A primeira linha da entrada contém um inteiro  $N$ , representando o número de retângulos na folha de papel. A segunda linha contém  $N$  inteiros  $A_i$ ,  $1 \leq i \leq N$ , representando a sequência de alturas dos retângulos.

## Saída

Seu programa deve imprimir uma linha contendo um inteiro representando o número máximo de pedaços possível, com um único corte horizontal.

## Restrições

- $2 \leq N \leq 10^5$
- $1 \leq A_i \leq 10^9$ , para  $1 \leq i \leq N$

## Informações sobre a pontuação

- Em um conjunto de casos de teste somando 40 pontos,  $N \leq 1000$

## Exemplo de entradas e saídas

### Entrada

5 6

1 2 4

1 3 3

4 3 6

4 5 2

2 4 1

3 5 5

### Saída

7



## Exemplo de entradas e saídas

### Entrada

7 10

1 2 5

3 1 32

1 4 3

2 3 4

2 6 20

6 3 1

6 4 9

6 5 6

3 7 18

5 7 2

### Saída

18

• a

## Solução $O(N \log N)$

```
1 // OBI 2017 - Nível Universitário: Fase 2
2 // Cortando o Papel (https://olimpiada.ic.unicamp.br/pratique/pu/2017/f2/papel/)
3 // Referência: https://noic.com.br/materiais-informatica/comentario-old/obi-2017/
4 #include <bits/stdc++.h>
5
6 using namespace std;
7 using ii = pair<int, int>;
8
9 int pieces(double x, int N, const vector<int>& hs)
10 {
11     auto L = 0, res = 0;
12
13     while (L < N)
14     {
15         if (hs[L] <= x)
16         {
17             ++L;
18             continue;
19         }
20
21         auto R = L + 1;
```

1. Dario e Xerxes
2. Frete
3. Mapa
4. Cortando o Papel
5. URI 1828 – Bazinga!