

Estruturas de Dados

Cristina Gomes Fernandes

ABB ótima

Suponha que tenhamos um **conjunto fixo de chaves** e que queremos repetidamente fazer testes de pertinência neste conjunto, ou seja, **buscas**.

Que estrutura de dados podemos usar neste caso?

ABB ótima

Suponha que tenhamos um **conjunto fixo de chaves** e que queremos repetidamente fazer testes de pertinência neste conjunto, ou seja, **buscas**.

Que estrutura de dados podemos usar neste caso?

Se não soubermos nada sobre as buscas que faremos, um **vetor ordenado e busca binária** podem ser uma boa opção.

ABB ótima

Suponha que tenhamos um **conjunto fixo de chaves** e que queremos repetidamente fazer testes de pertinência neste conjunto, ou seja, **buscas**.

Que estrutura de dados podemos usar neste caso?

Se não soubermos nada sobre as buscas que faremos, um **vetor ordenado e busca binária** podem ser uma boa opção.

No entanto, há situações em que temos alguma informação estatística sobre as buscas. Por exemplo, podemos ter a **freqüência estimada de busca de cada chave**.

Como podemos levar essa informação extra em conta para derivar uma ED melhor do que o vetor ordenado com a busca binária?

Exemplo

$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

Qual é o número esperado de comparações que seriam feitas em um vetor ordenado com a busca binária?

Exemplo

$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

Qual é o número esperado de comparações que seriam feitas em um vetor ordenado com a busca binária?

$$\frac{1}{7} \cdot 2 + \frac{2}{7} \cdot 1 + \frac{4}{7} \cdot 2 = \frac{12}{7}$$

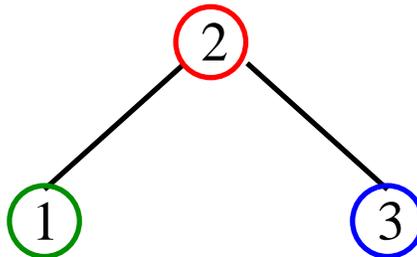
Exemplo

$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

Qual é o número esperado de comparações que seriam feitas em um vetor ordenado com a busca binária?

$$\frac{1}{7} \cdot 2 + \frac{2}{7} \cdot 1 + \frac{4}{7} \cdot 2 = \frac{12}{7}$$

Isso é também o número esperado de comparações feitas na seguinte ABB:



Exemplo

$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

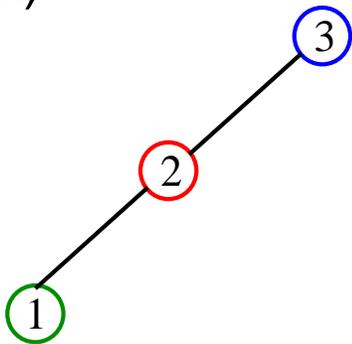
Uma outra ABB não seria melhor, dadas as frequências?

Exemplo

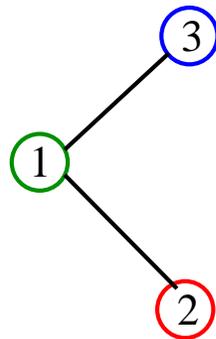
$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

Uma outra ABB não seria melhor, dadas as freqüências?

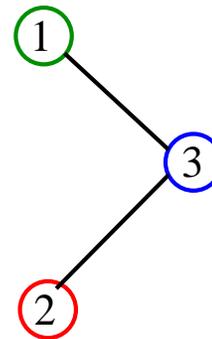
(a)



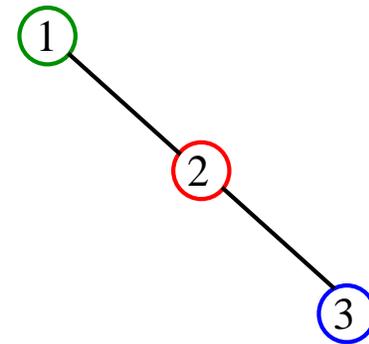
(b)



(c)



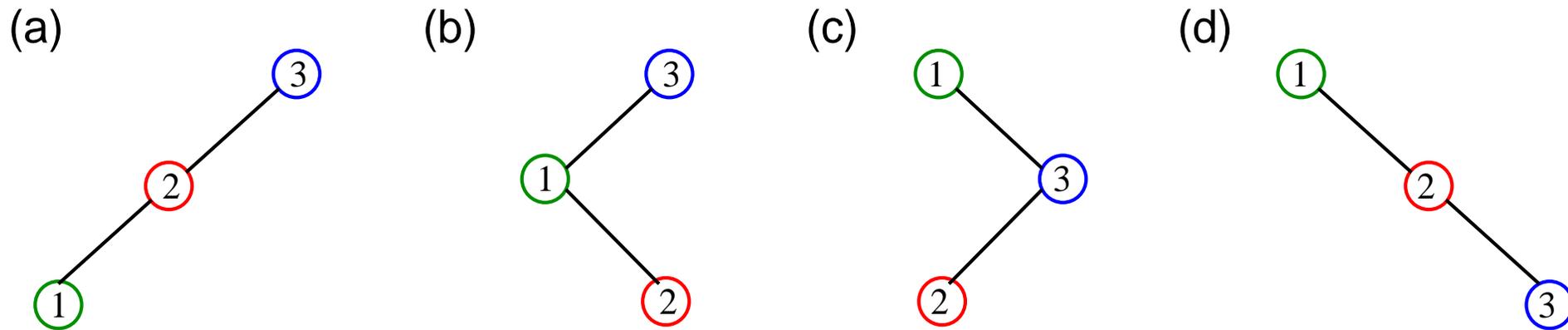
(d)



Exemplo

$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

Uma outra ABB não seria melhor, dadas as freqüências?



$$(a) \quad \frac{1}{7} \cdot 3 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 1 = \frac{11}{7}$$

$$(b) \quad \frac{1}{7} \cdot 2 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 1 = \frac{12}{7}$$

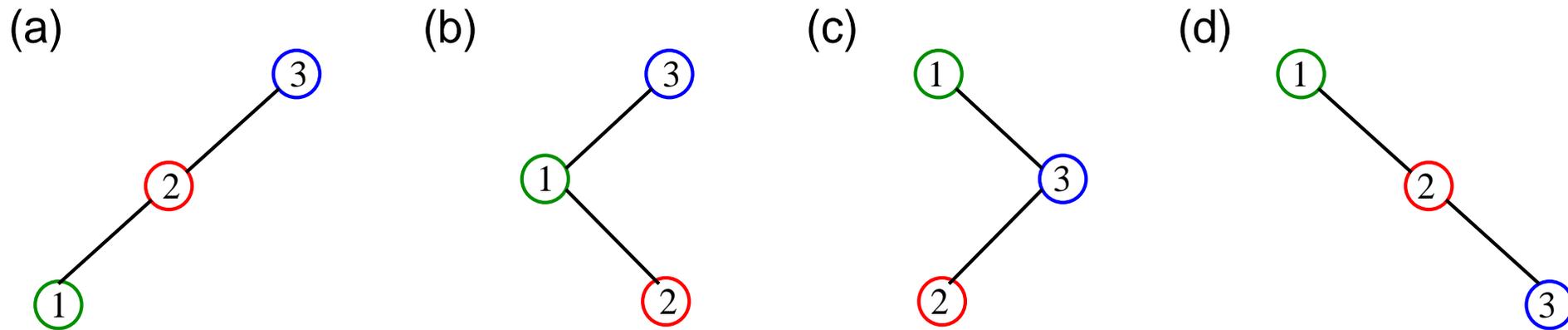
$$(c) \quad \frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 2 = \frac{15}{7}$$

$$(d) \quad \frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 3 = \frac{17}{7}$$

Exemplo

$$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$$

Uma outra ABB não seria melhor, dadas as freqüências?



$$(a) \quad \frac{1}{7} \cdot 3 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 1 = \frac{11}{7}$$

$$(b) \quad \frac{1}{7} \cdot 2 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 1 = \frac{12}{7}$$

$$(c) \quad \frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 2 = \frac{15}{7}$$

$$(d) \quad \frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 3 = \frac{17}{7}$$

← ABB ótima neste caso!

ABB ótima

Problema: Dado n e uma distribuição de probabilidades (d.p.) p_1, p_2, \dots, p_n , encontrar uma ABB ótima para esta d.p..

ABB ótima

Problema: Dado n e uma distribuição de probabilidades (d.p.) p_1, p_2, \dots, p_n , encontrar uma ABB ótima para esta d.p..

Ou seja, encontrar uma ABB para as chaves $1, 2, \dots, n$ cujo número esperado de comparações para uma busca com d.p. p_1, p_2, \dots, p_n é mínimo.

ABB ótima

Problema: Dado n e uma distribuição de probabilidades (d.p.) p_1, p_2, \dots, p_n , encontrar uma ABB ótima para esta d.p..

Ou seja, encontrar uma ABB para as chaves $1, 2, \dots, n$ cujo número esperado de comparações para uma busca com d.p. p_1, p_2, \dots, p_n é mínimo.

Seja $e[i, j]$ o número esperado mínimo de comparações para uma busca em uma ABB com as chaves $i, i + 1, \dots, j$, onde a busca segue a d.p. p_i, p_{i+1}, \dots, p_j .

Vale a seguinte recorrência para $e[i, j]$.

ABB ótima

Vale a seguinte recorrência para $e[i, j]$:

$$e[i, j] = \begin{cases} 0 & \text{se } i > j \\ \min_{i \leq k \leq j} \{e[i, k-1] + e[k+1, j]\} + w[i, j] & \text{se } i \leq j \end{cases}$$

onde

$$w[i, j] = \begin{cases} 0 & \text{se } i > j \\ w[i, j-1] + p_j & \text{se } i \leq j, \end{cases}$$

ou seja, $w[i, j] = \sum_{k=i}^j p_k$.

Podemos então usar PD para calcular $e[i, j]$ e $w[i, j]$.

ABB ótima

ABBÓTIMA (p, n)

```
1  para  $i \leftarrow 1$  até  $n + 1$  faça
2     $w[i, i - 1] \leftarrow 0$ 
3     $e[i, i - 1] \leftarrow 0$ 
4  para  $\ell \leftarrow 1$  até  $n$  faça
5    para  $i \leftarrow 1$  até  $n - \ell + 1$  faça
6       $j \leftarrow i + \ell - 1$ 
7       $w[i, j] \leftarrow w[i, j - 1] + p[j]$ 
8       $e[i, j] \leftarrow \infty$ 
9      para  $k \leftarrow i$  até  $j$  faça
10          $aux \leftarrow e[i, k - 1] + e[k + 1, j] + w[i, j]$ 
11         se  $aux < e[i, j]$ 
12           então  $e[i, j] \leftarrow aux$ 
13              $r[i, j] \leftarrow k$ 
14 devolva ( $e, r$ )
```

ABB ótima

Exercício: Escreva uma rotina **MontaABB**(r, n) que receba como parâmetro a matriz $r_{n \times n}$ devolvida pelo **ABBÓTIMA**(p, n) e devolva um apontador para a raiz de uma ABB ótima para $p[1..n]$.

ABB ótima: caso geral

No que segue, considere que a 0-ésima chave é $-\infty$ e que a $(n + 1)$ -ésima chave é $+\infty$.

Entrada do caso geral: d.p. $q_0, p_1, q_1, \dots, q_{n-1}, p_n, q_n$, onde p_i é a probabilidade de se buscar a i -ésima chave, e q_i é a probabilidade de se fazer uma busca mal-sucedida por um x entre a i -ésima e a $(i + 1)$ -ésima chave.

Saída do caso geral: ABB ótima, levando em consideração na esperança as buscas mal-sucedidas.

Exercício: Resolva o caso geral da ABB ótima.

Dica: Redefina $e[i, j]$ como o valor de uma ABB ótima para buscas de valores estritamente maiores que a chave $i - 1$ e estritamente menores que a chave $i + 1$.

Ajuste a recorrência e o algoritmo adequadamente para essa nova definição.