Análise amortizada

Sec 17.4 Tabelas dinâmicas do CLRS

Parte das notas de aula de um curso do Robert Tarjan.

"*Amortized Analysis Explained*"

por Rebecca Fiebrink, Princeton University

Análise amortizada

Serve para analisar uma sequência de operações ou iterações onde o pior caso individual não reflete o pior caso da sequência.

Em outras palavras, serve para melhorar análises de pior caso que baseiem-se diretamente no pior caso de uma operação/iteração e que deem uma delimitação frouxa para o tempo de pior caso da sequência.

Métodos:

- agregado
- por créditos
- potencial

Aula passada

Exemplos de análise amortizada:

- odômetro binário
  - análise agregada
  - análise por créditos
  - análise com função potencial
- pilha com operação única
  - análise agregada
  - análise por créditos
  - análise com função potencial
- tabelas dinâmicas
  - análise agregada
  - análise por créditos
  - **análise com função potencial**

Tabelas dinâmicas

Vetor que sofre inserções. Cada inserção custa 1.

Inicialmente o vetor tem 0 posições.

Na primeira inserção, um vetor com uma posição é alocado, e o item em questão é inserido.

A cada inserção em que o vetor está cheio, antes da inserção propriamente dita, um vetor do dobro do tamanho é alocado, o vetor anterior é copiado para o novo vetor e depois é desalocado.

O custo no pior caso de uma inserção é alto, pois pode haver uma realocação.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$,

$$c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais 1} \\ i & \text{se } i \text{ é potência de 2 mais 1.} \end{cases}$$

Chame de velho um item que já estava no vetor T
no momento da última realocação do vetor T,
e de novos os itens inseridos após a última realocação.

Método potencial:

Que função potencial você usaria neste caso?

Tome $\Phi(T)$ como duas vezes o número de elementos novos em T.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$,

$$c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais 1} \\ i & \text{se } i \text{ é potência de 2 mais 1.} \end{cases}$$

Chame de velho um item que já estava no vetor T
no momento da última realocação do vetor T,
e de novos os itens inseridos após a última realocação.

Método potencial:

Que função potencial você usaria neste caso?

Tome $\Phi(T)$ como duas vezes o número de elementos novos em T.

T_i: estado do vetor T imediatamente após a inserção i

Note que $\Phi(T_0) = 0$ e $\Phi(T_i) \geq 0$.

Custo por inserção e função potencial

Para $i = 1, 2, \ldots, n$,

$$c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais 1} \\ i & \text{se } i \text{ é potência de 2 mais 1.} \end{cases}$$

Chame de velho um item que já estava no vetor T
no momento da última realocação do vetor T,
e de novos os itens inseridos após a última realocação.

Método potencial:

Tome $\Phi(T) = 2(\mathrm{num}(T) - \mathrm{size}(T)/2)$
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Custo por inserção e função potencial

Para $i = 1, 2, \ldots, n$,

$$c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais 1} \\ i & \text{se } i \text{ é potência de 2 mais 1.} \end{cases}$$

Chame de velho um item que já estava no vetor T
no momento da última realocação do vetor T,
e de novos os itens inseridos após a última realocação.

Método potencial:

Tome $\Phi(T) = 2(\mathrm{num}(T) - \mathrm{size}(T)/2) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Custo por inserção e função potencial

Para $i = 1, 2, \ldots, n$,

$$c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais 1} \\ i & \text{se } i \text{ é potência de 2 mais 1.} \end{cases}$$

Chame de velho um item que já estava no vetor T
no momento da última realocação do vetor T,
e de novos os itens inseridos após a última realocação.

Método potencial:

Tome $\Phi(T) = 2(\mathrm{num}(T) - \mathrm{size}(T)/2) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Vamos calcular $\hat{c}_i = c_i + \Phi(T_i) - \Phi(T_{i-1})$.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n, \;\; c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Vamos calcular $\hat{c}_i = c_i + \Phi(T_i) - \Phi(T_{i-1})$.

Dois casos:

- i não é potência de 2 mais um.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Vamos calcular $\hat{c}_i = c_i + \Phi(T_i) - \Phi(T_{i-1})$.

Dois casos:

- i não é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1$ e $\mathrm{size}(T_i) = \mathrm{size}(T_{i-1})$.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Vamos calcular $\hat{c}_i = c_i + \Phi(T_i) - \Phi(T_{i-1})$.

Dois casos:

- i não é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1$ e $\mathrm{size}(T_i) = \mathrm{size}(T_{i-1})$.

$$\begin{aligned} \hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\ &= 1 + (2\,\mathrm{num}(T_i) - \mathrm{size}(T_i)) - (2\,\mathrm{num}(T_{i-1}) - \mathrm{size}(T_{i-1})) \\ &= 1 + 2 + 0 = 3. \end{aligned}$$

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $\quad c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n, \;\; c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1 = c_i$ e
  $\mathrm{size}(T_i) = 2\,\mathrm{size}(T_{i-1}) = 2\,\mathrm{num}(T_{i-1})$

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1 = c_i$ e
  $\mathrm{size}(T_i) = 2\,\mathrm{size}(T_{i-1}) = 2\,\mathrm{num}(T_{i-1}) = 2(c_i - 1)$.

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1 = c_i$ e
  $\mathrm{size}(T_i) = 2\,\mathrm{size}(T_{i-1}) = 2\,\mathrm{num}(T_{i-1}) = 2(c_i - 1)$.

$$\begin{aligned} \hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\ &= c_i + (2\,\mathrm{num}(T_i) - \mathrm{size}(T_i)) - (2\,\mathrm{num}(T_{i-1}) - \mathrm{size}(T_{i-1})) \\ &= c_i + (2c_i - 2(c_i - 1)) \end{aligned}$$

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1 = c_i$ e
  $\mathrm{size}(T_i) = 2\,\mathrm{size}(T_{i-1}) = 2\,\mathrm{num}(T_{i-1}) = 2(c_i - 1)$.

$$\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= c_i + (2\,\mathrm{num}(T_i) - \mathrm{size}(T_i)) - (2\,\mathrm{num}(T_{i-1}) - \mathrm{size}(T_{i-1})) \\
&= c_i + (2c_i - 2(c_i - 1)) - (2(c_i - 1) - (c_i - 1))
\end{aligned}$$

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$, $c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um.
  $\mathrm{num}(T_i) = \mathrm{num}(T_{i-1}) + 1 = c_i$ e
  $\mathrm{size}(T_i) = 2\,\mathrm{size}(T_{i-1}) = 2\,\mathrm{num}(T_{i-1}) = 2(c_i - 1)$.

$$
\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= c_i + (2\,\mathrm{num}(T_i) - \mathrm{size}(T_i)) - (2\,\mathrm{num}(T_{i-1}) - \mathrm{size}(T_{i-1})) \\
&= c_i + (2c_i - 2(c_i - 1)) - (2(c_i - 1) - (c_i - 1)) \\
&= c_i + 2 - (c_i - 1) \;=\; 3.
\end{aligned}
$$

Tabelas dinâmicas: método potencial

Para $i = 1, 2, \ldots, n$,

$$c_i = \begin{cases} 1 & \text{se } i \text{ não é potência de 2 mais um} \\ i & \text{se } i \text{ é potência de 2 mais um.} \end{cases}$$

Tome $\Phi(T) = 2\,\mathrm{num}(T) - \mathrm{size}(T)$,
onde $\mathrm{num}(T)$ é o número de elementos em T e
$\mathrm{size}(T)$ é o tamanho de T.

Dois casos:

- i não é potência de 2 mais um: $\hat{c}_i = 3$.
- i é potência de 2 mais um: $\hat{c}_i = 3$.

Conclusão: custo amortizado por inserção é 3.

Move to front

Considere uma lista ligada.

Custo do acesso ao i-ésimo elemento: i.

Move to front

Considere uma lista ligada.

Custo do acesso ao i-ésimo elemento: i.

Custo de trocar dois elementos consecutivos de lugar:
constante, descontado o tempo de acessar um deles.

Move to front

Considere uma lista ligada.

Custo do acesso ao i-ésimo elemento: i.

Custo de trocar dois elementos consecutivos de lugar:
constante, descontado o tempo de acessar um deles.

Heurística MTF:
ao acessar o elemento x, mova-o para o ínicio da lista.

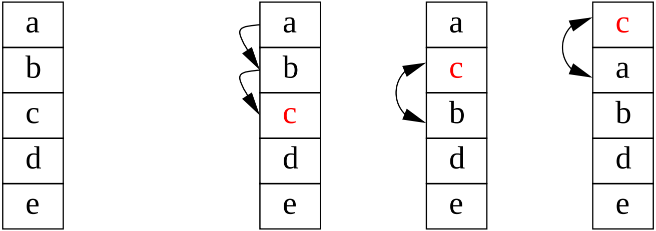
Move to front

Considere uma lista ligada.

Custo do acesso ao i-ésimo elemento: i.

Custo de trocar dois elementos consecutivos de lugar:
constante, descontado o tempo de acessar um deles.

Heurística MTF:
ao acessar o elemento x, mova-o para o ínicio da lista.



Move to front

Considere uma lista ligada.

Custo do acesso ao i-ésimo elemento: i.

Custo de trocar dois elementos consecutivos de lugar:
constante, descontado o tempo de acessar um deles.

Heurística MTF:
ao acessar o elemento x, mova-o para o ínicio da lista.

Com MTF, custo do acesso ao i-ésimo elemento: $2i - 1$.

Move to front

Considere uma lista ligada.

Custo do acesso ao i-ésimo elemento: i.

Custo de trocar dois elementos consecutivos de lugar:
constante, descontado o tempo de acessar um deles.

Heurística MTF:
ao acessar o elemento x, mova-o para o ínicio da lista.

Com MTF, custo do acesso ao i-ésimo elemento: $2i - 1$.

Considere uma sequência de acessos à lista.

Qual é o custo de MTF se comparado
com um algoritmo ótimo de acesso?

Move to front

Lista ligada e sequência de acessos a ela.

Heurística MTF:
ao acessar o elemento x, mova-o para o ínicio da lista.

Acesso ao i-ésimo elemento custa $2i - 1$.

Qual é o custo de MTF se comparado
com um algoritmo ótimo de acesso?

Move to front

Lista ligada e sequência de acessos a ela.

Heurística MTF:
ao acessar o elemento x, mova-o para o ínicio da lista.

Acesso ao i-ésimo elemento custa $2i - 1$.

Qual é o custo de MTF se comparado
com um algoritmo ótimo de acesso?

A: algoritmo arbitrário de acesso à lista.

Análise amortizada:

$$\text{custo do MTF} \leq 4 \text{ vezes o custo de } A.$$

Operações sobre a lista

Acesso e troca com anterior.

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

Operações sobre a lista

Acesso e troca com anterior.

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

Seja x o elemento acessado.

Operações sobre a lista

Acesso e troca com anterior.

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

Seja x o elemento acessado.

Seja k a posição de x na lista de MTF.

Seja i a posição de x na lista de A.

Operações sobre a lista

Acesso e troca com anterior.

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

Seja x o elemento acessado.

Seja k a posição de x na lista de MTF.

Seja i a posição de x na lista de A.

Suponha que A não troca ninguém de lugar.

Custo pelo acesso a x por MTF: $2k - 1$.

Custo pelo acesso a x por A: i.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões
da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Em MTF, depois do acesso,
pares com x e cada um dos $k-1$ elementos trocaram de posição.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Em MTF, depois do acesso, pares com x e cada um dos $k-1$ elementos trocaram de posição.

Existem $i-1$ elementos na lista de A na frente de x.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Em MTF, depois do acesso, pares com x e cada um dos $k-1$ elementos trocaram de posição.

Existem $i-1$ elementos na lista de A na frente de x.

$\leq \min\{k-1, i-1\}$ inversões a mais depois do acesso

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões
da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Em MTF, depois do acesso,
pares com x e cada um dos $k-1$ elementos trocaram de posição.

Existem $i-1$ elementos na lista de A na frente de x.

$\leq \min\{k-1, i-1\}$ inversões a mais depois do acesso

$\geq (k-1) - \min\{k-1, i-1\}$ inversões a menos

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões
da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Em MTF, depois do acesso,
pares com x e cada um dos $k-1$ elementos trocaram de posição.

Existem $i-1$ elementos na lista de A na frente de x.

$\leq \min\{k-1, i-1\}$ inversões a mais depois do acesso

$\geq (k-1) - \min\{k-1, i-1\}$ inversões a menos

Então
$\Delta\Phi \leq 2(2\min\{k-1, i-1\} - (k-1))$

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

k: posição de x na lista de MTF.
i: posição de x na lista de A.

Em MTF, depois do acesso,
pares com x e cada um dos $k-1$ elementos trocaram de posição.

Existem $i-1$ elementos na lista de A na frente de x.

$\leq \min\{k-1, i-1\}$ inversões a mais depois do acesso

$\geq (k-1) - \min\{k-1, i-1\}$ inversões a menos

Então
$\Delta\Phi \leq 2(2\min\{k-1, i-1\} - (k-1)) = 4\min\{k-1, i-1\} - 2(k-1)$.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

$\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1)$.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

$\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1)$.

Custo amortizado:

$$\begin{aligned}
\hat{c} &= c + \Delta\Phi \\
&\leq 2k - 1 + 4\min\{k-1, i-1\} - 2(k-1) \\
&\leq 1 + 4\min\{k-1, i-1\} \\
&\leq 4i.
\end{aligned}$$

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

$\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1)$.

Custo amortizado:

$$
\begin{aligned}
\hat{c} &= c + \Delta\Phi \\
&\leq 2k - 1 + 4\min\{k-1, i-1\} - 2(k-1) \\
&\leq 1 + 4\min\{k-1, i-1\} \\
&\leq 4i.
\end{aligned}
$$

Lembre-se que o custo de A é i.

Custo amortizado do acesso

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

$\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1)$.

Custo amortizado:

$$
\begin{aligned}
\hat{c} &= c + \Delta\Phi \\
&\leq 2k - 1 + 4\min\{k-1, i-1\} - 2(k-1) \\
&\leq 1 + 4\min\{k-1, i-1\} \\
&\leq 4i.
\end{aligned}
$$

Lembre-se que o custo de A é i.

Logo o custo amortizado por acesso de A é no máximo 4.

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

E se A fizer trocas também?

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

E se A fizer trocas também?

Com uma troca, o custo de A sobe de 1.

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

E se A fizer trocas também?

Com uma troca, o custo de A sobe de 1.

E o potencial, como se altera? Sobe ou desce de 2.

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

E se A fizer trocas também?

Com uma troca, o custo de A sobe de 1.

E o potencial, como se altera? Sobe ou desce de 2.

Ou seja, o custo de A é $i + t$,
onde t é o número de trocas de A após o acesso de x,
e $\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1) + 2t$.

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

E se A fizer trocas também?

Com uma troca, o custo de A sobe de 1.

E o potencial, como se altera? Sobe ou desce de 2.

Ou seja, o custo de A é $i + t$,
onde t é o número de trocas de A após o acesso de x,
e $\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1) + 2t$.

Custo amortizado: $\hat{c} \leq 4i + 2t \leq 4(i + t)$.

Custo amortizado

Função potencial Φ: 2 vezes o número de inversões da lista de MTF em relação à lista de A.

O custo amortizado por acesso é no máximo 4 em relação ao custo de A... se A não fizer trocas...

E se A fizer trocas também?

Com uma troca, o custo de A sobe de 1.

E o potencial, como se altera? Sobe ou desce de 2.

Ou seja, o custo de A é $i + t$,
onde t é o número de trocas de A após o acesso de x,
e $\Delta\Phi \leq 4\min\{k-1, i-1\} - 2(k-1) + 2t$.

Custo amortizado: $\hat{c} \leq 4i + 2t \leq 4(i + t)$.

Custo amortizado por operação de A é no máximo 4.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

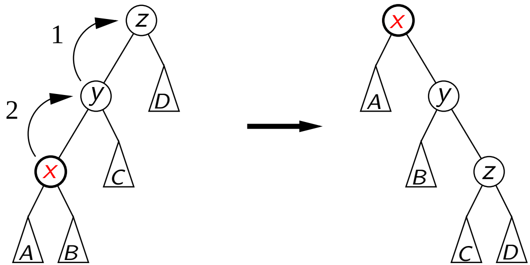
As rotações são feitas numa ordem específica, de duas em duas.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

As rotações são feitas numa ordem específica, de duas em duas.



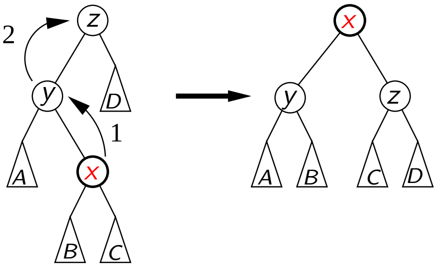
Acima, o rr splay step.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

As rotações são feitas numa ordem específica, de duas em duas.



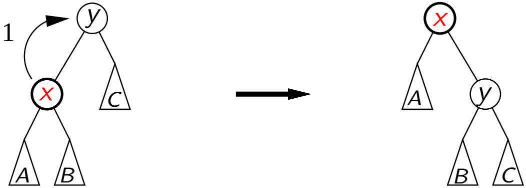
Acima, o lr splay step.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

As rotações são feitas numa ordem específica, de duas em duas.
Eventualmente no final uma rotação simples é necessária.



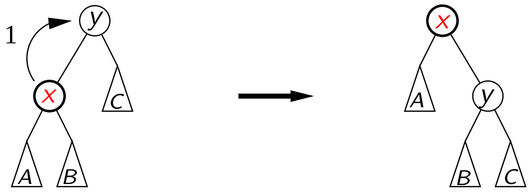
Acima, o r splay step.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

As rotações são feitas numa ordem específica, de duas em duas.
Eventualmente no final uma rotação simples é necessária.



Acima, o r splay step.

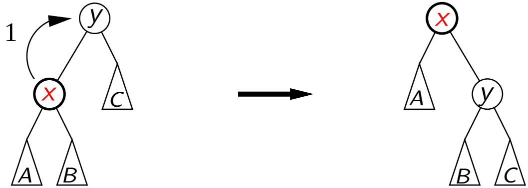
Além destes, o l splay, o rl splay e o ll splay step.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

As rotações são feitas numa ordem específica, de duas em duas.
Eventualmente no final uma rotação simples é necessária.



Acima, o r splay step.

Além destes, o l splay, o rl splay e o ll splay step.

Splay steps são realizados até que x seja raiz.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

Custo de pior caso do SPLAY: $\Theta(n)$,
onde n é o número de elementos na splay tree.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

Custo de pior caso do SPLAY: $\Theta(n)$,
onde n é o número de elementos na splay tree.

Queremos mostrar que uma splay tree tem
comportamento mais parecido com o de uma ABBB.

Splay trees

ABB: árvore binária de busca

Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

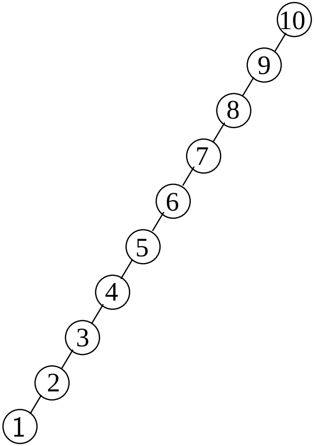
Custo de pior caso do SPLAY: $\Theta(n)$,
onde n é o número de elementos na splay tree.

Queremos mostrar que uma splay tree tem
comportamento mais parecido com o de uma ABBB.

(ABBB: ABB balanceada)

Splay trees: pior caso

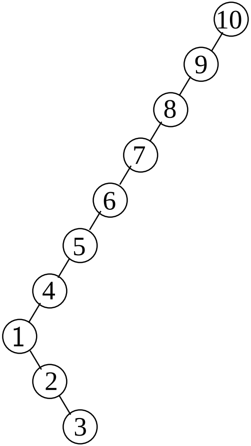
SPLAY$(1, S)$



Splay trees: pior caso

$\textsc{Splay}(1, S)$

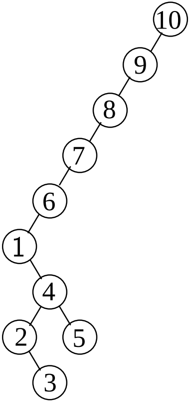
primeiro splay step



Splay trees: pior caso

$\textsc{Splay}(1, S)$

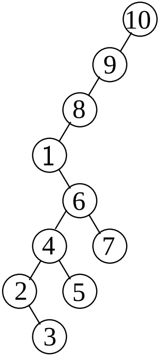
segundo splay step



Splay trees: pior caso

$\text{SPLAY}(1, S)$

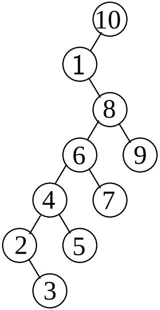
terceiro splay step



Splay trees: pior caso

$\textsc{Splay}(1, S)$

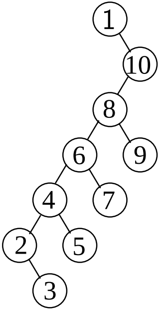
quarto splay step



Splay trees: pior caso

SPLAY$(1, S)$

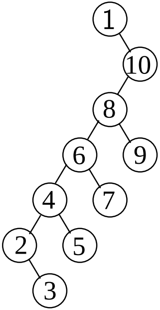
quinto splay step



Splay trees: pior caso

SPLAY$(1, S)$

quinto splay step



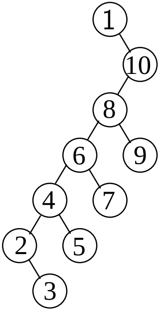
Total de rotações: 9

Em geral, esse caso é $\Theta(n)$, onde n é o número de nós.

Depois disso, o maior custo de um SPLAY nesta árvore é 6.

Splay trees: mais um exemplo

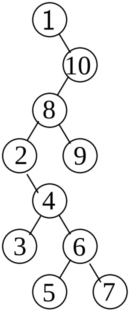
SPLAY$(2, S)$



Splay trees: mais um exemplo

SPLAY$(2, S)$

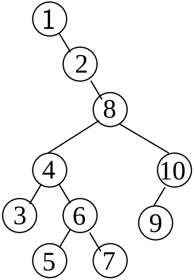
primeiro splay step



Splay trees: mais um exemplo

SPLAY$(2, S)$

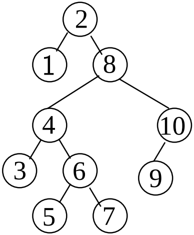
segundo splay step



Splay trees: mais um exemplo

SPLAY$(2, S)$

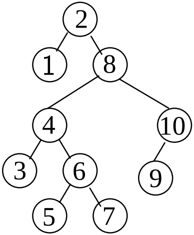
terceiro splay step



Splay trees: mais um exemplo

SPLAY$(2, S)$

terceiro splay step



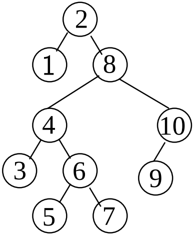
Total de rotações: 5

Árvore bem mais balanceada após estes dois SPLAYs.

Splay trees: mais um exemplo

SPLAY$(2, S)$

terceiro splay step



Total de rotações: 5

Árvore bem mais balanceada após estes dois SPLAYs.

Custo: número de rotações.

Jorge Stolfi