Análise amortizada

Notas de aula de um curso do Robert Tarjan

"*Amortized Analysis Explained*"

por Rebecca Fiebrink, Princeton University

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.

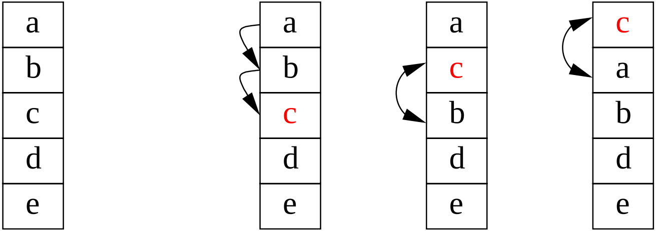
Move to front

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:

custo do MTF $\leq$ 4 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 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 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 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 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 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.

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

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:

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

Logo o custo amortizado por custo de 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 é menor que 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 é menor que 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 é menor que 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 é menor que 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 é menor que 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 é menor que 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 é menor que 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 ≤ 4.

Splay trees

São árvores binárias de busca auto-ajustáveis!

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.

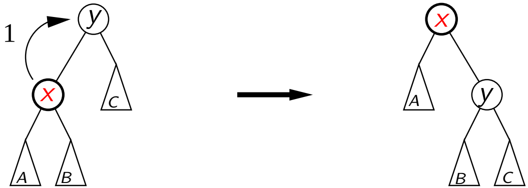
Splay trees

São árvores binárias de busca auto-ajustáveis!

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.

Lembram-se das rotações usadas para balanceamento?



Splay trees

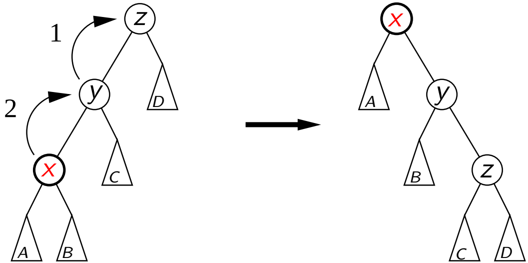
Operação SPLAY(x, S), onde S é uma splay tree:
quando x é acessado, move-se x para a raiz por rotações.

Para a análise, olhamos duas rotações de cada vez.

Splay trees

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

Para a análise, olhamos duas rotações de cada vez.

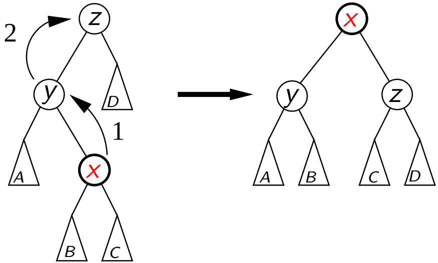


Acima, o rr splay step.

Splay trees

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

Para a análise, olhamos duas rotações de cada vez.

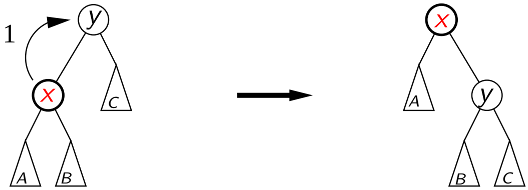


Acima, o lr splay step.

Splay trees

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

Para a análise, olhamos duas rotações de cada vez.
Eventualmente no final uma rotação simples é necessária.

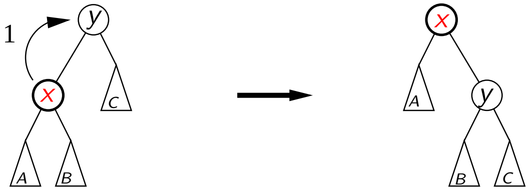


Acima, o r splay step.

Splay trees

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

Para a análise, olhamos duas rotações de cada vez.
Eventualmente no final uma rotação simples é necessária.



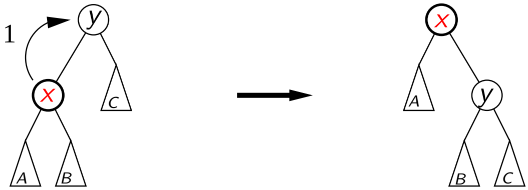
Acima, o r splay step.

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

Splay trees

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

Para a análise, olhamos duas rotações de cada vez.
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

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

Splay trees

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

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
um comportamento mais parecido com o de uma ABBB.

Splay trees

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

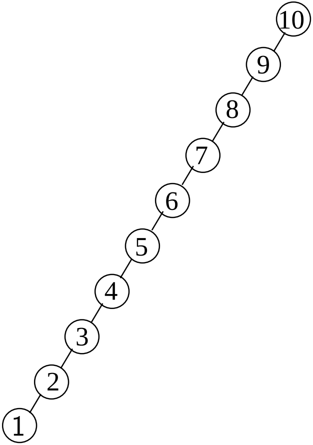
Custo de pior caso do SPLAY: $\Theta(n)$,
onde n é o número de elementos na splay tree.

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

(ABBB: ABB balanceada)

Splay trees: pior caso

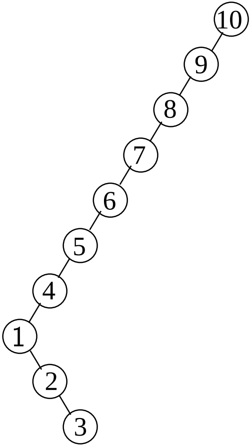
SPLAY$(1, S)$



Splay trees: pior caso

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

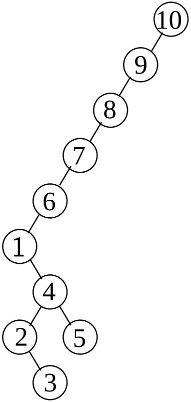
primeiro splay step



Splay trees: pior caso

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

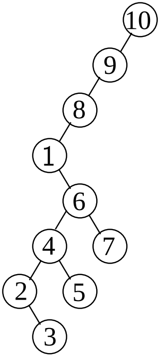
segundo splay step



Splay trees: pior caso

SPLAY$(1, S)$

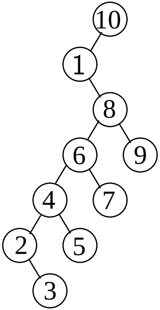
terceiro splay step



Splay trees: pior caso

SPLAY$(1, S)$

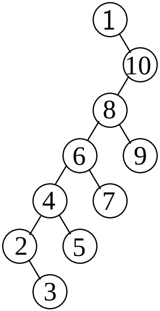
quarto splay step



Splay trees: pior caso

SPLAY$(1, S)$

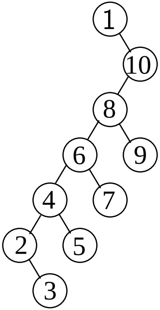
quinto splay step



Splay trees: pior caso

SPLAY$(1, S)$

quinto splay step



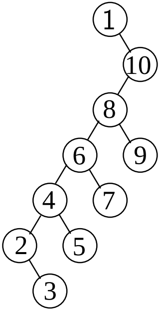
Total de rotações: 9

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

Agora, o maior custo de um SPLAY nesta árvore é 7.

Splay trees: mais um exemplo

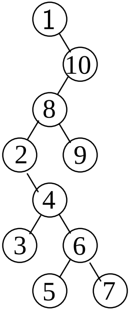
SPLAY$(2, S)$



Splay trees: mais um exemplo

SPLAY$(2, S)$

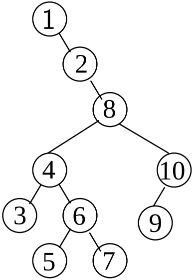
primeiro splay step



Splay trees: mais um exemplo

$\text{SPLAY}(2, S)$

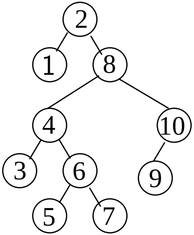
segundo splay step



Splay trees: mais um exemplo

SPLAY$(2, S)$

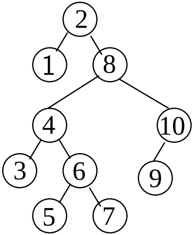
terceiro splay step



Splay trees: mais um exemplo

SPLAY$(2, S)$

terceiro splay step



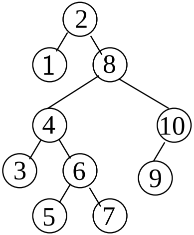
Total de rotações: 5

Árvore bem mais balanceada após estes dois SPLAYs.

Splay trees: mais um exemplo

SPLAY$(2, S)$

terceiro splay step



Total de rotações: 5

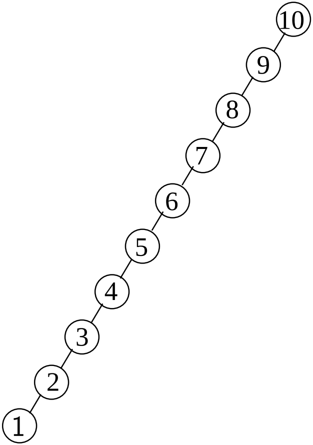
Árvore bem mais balanceada após estes dois SPLAYs.

Custo: número de rotações.

Splay trees: por que steps duplos?

O que acontece se implementarmos o SPLAY com apenas os splay simples?

Como ficaria o resultado de SPLAY$(1, S)$ nessa árvore?



Splay trees

S: splay tree

Splay trees

S: splay tree

$S_i(x)$: subárvore de S enraizada em x no instante i
$s_i(x) = |S_i(x)|$

Splay trees

S: splay tree

$S_i(x)$: subárvore de S enraizada em x no instante i
$s_i(x) = |S_i(x)|$

Tome $r_i(x) = \lg s_i(x)$.

$r_i(x)$ indica o potencial local no nó x.

Splay trees

S: splay tree

$S_i(x)$: subárvore de S enraizada em x no instante i
$s_i(x) = |S_i(x)|$

Tome $r_i(x) = \lg s_i(x)$.

$r_i(x)$ indica o potencial local no nó x.

Seja $\Phi_i = \sum_x r_i(x)$.

Splay trees

S: splay tree

$S_i(x)$: subárvore de S enraizada em x no instante i
$s_i(x) = |S_i(x)|$

Tome $r_i(x) = \lg s_i(x)$.

$r_i(x)$ indica o potencial local no nó x.

Seja $\Phi_i = \sum_x r_i(x)$.

Mostraremos que o custo amortizado por SPLAY é $\mathrm{O}(\lg n)$.

Splay trees

S: splay tree

$S_i(x)$: subárvore de S enraizada em x no instante i
$s_i(x) = |S_i(x)|$

Tome $r_i(x) = \lg s_i(x)$.

$r_i(x)$ indica o potencial local no nó x.

Seja $\Phi_i = \sum_x r_i(x)$.

Mostraremos que o custo amortizado por SPLAY é $\mathrm{O}(\lg n)$.

Análise amortizada dos splay steps.

Splay trees

S: splay tree

$S_i(x)$: subárvore de S enraizada em x no instante i
$s_i(x) = |S_i(x)|$

Tome $r_i(x) = \lg s_i(x)$.

$r_i(x)$ indica o potencial local no nó x.

Seja $\Phi_i = \sum_x r_i(x)$.

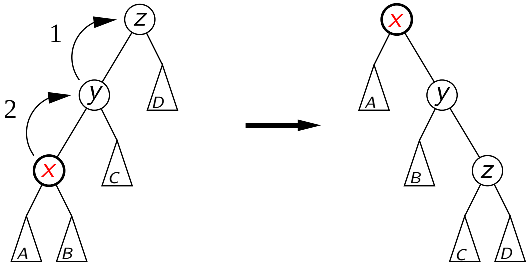
Mostraremos que o custo amortizado por SPLAY é $\mathrm{O}(\lg n)$.

Análise amortizada dos splay steps.

x participa de todos os splay steps de SPLAY(x, S).

Análise amortizada dos splay steps

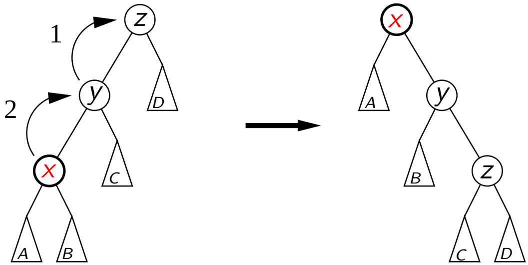
Caso do rr splay step.



Custo real: 2

Análise amortizada dos splay steps

Caso do rr splay step.

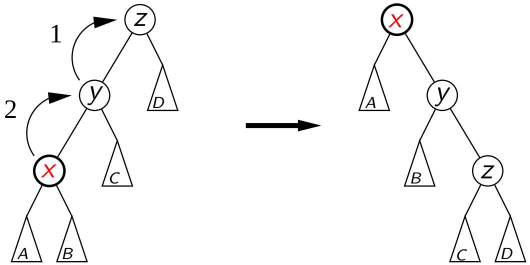


Custo real: 2

Alteração no potencial:
$\Phi_i - \Phi_{i-1} = \sum_w r_i(w) - \sum_w r_{i-1}(w) = \sum_w (r_i(w) - r_{i-1}(w))$

Análise amortizada dos splay steps

Caso do rr splay step.



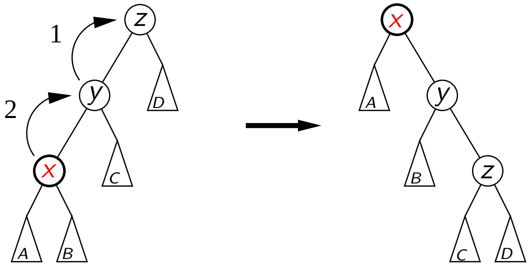
Custo real: 2

Alteração no potencial:

$$\Phi_i - \Phi_{i-1} = \sum_w r_i(w) - \sum_w r_{i-1}(w) = \sum_w (r_i(w) - r_{i-1}(w))$$
$$= (r_i(x) - r_{i-1}(x)) + (r_i(y) - r_{i-1}(y)) + (r_i(z) - r_{i-1}(z))$$

Análise amortizada dos splay steps

Caso do rr splay step.



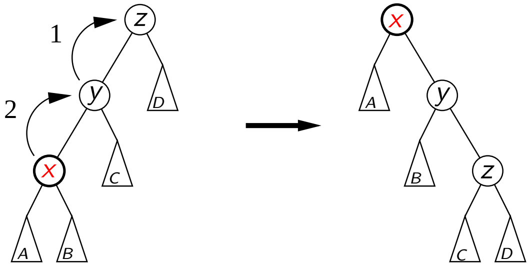
Custo real: 2

Alteração no potencial:

$$
\begin{aligned}
\Phi_i - \Phi_{i-1} &= \sum_w r_i(w) - \sum_w r_{i-1}(w) = \sum_w (r_i(w) - r_{i-1}(w)) \\
&= (r_i(x) - r_{i-1}(x)) + (r_i(y) - r_{i-1}(y)) + (r_i(z) - r_{i-1}(z)) \\
&= (r_i(y) + r_i(z)) - (r_{i-1}(x) + r_{i-1}(y))
\end{aligned}
$$

Análise amortizada dos splay steps

Caso do rr splay step.



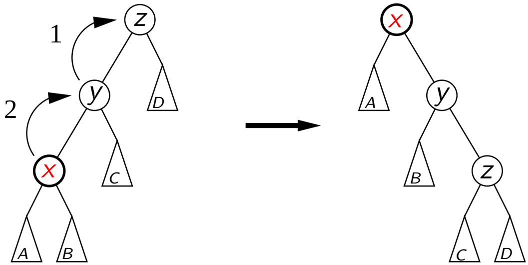
Custo real: 2

Alteração no potencial:

$$
\begin{aligned}
\Phi_i - \Phi_{i-1} &= \sum_w r_i(w) - \sum_w r_{i-1}(w) = \sum_w (r_i(w) - r_{i-1}(w)) \\
&= (r_i(x) - r_{i-1}(x)) + (r_i(y) - r_{i-1}(y)) + (r_i(z) - r_{i-1}(z)) \\
&= (r_i(y) + r_i(z)) - (r_{i-1}(x) + r_{i-1}(y)) \\
&< r_i(y) + r_i(z) - 2\, r_{i-1}(x)
\end{aligned}
$$

Análise amortizada dos splay steps

Caso do rr splay step.



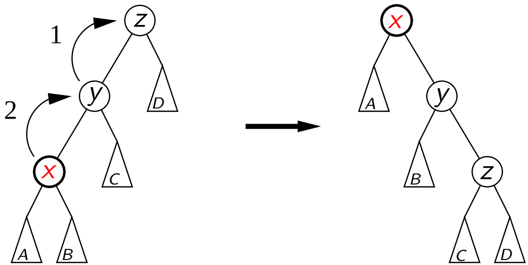
Custo real: 2

Alteração no potencial:

$$\begin{aligned}\Phi_i - \Phi_{i-1} &= \sum_w r_i(w) - \sum_w r_{i-1}(w) = \sum_w (r_i(w) - r_{i-1}(w)) \\ &= (r_i(x) - r_{i-1}(x)) + (r_i(y) - r_{i-1}(y)) + (r_i(z) - r_{i-1}(z)) \\ &= (r_i(y) + r_i(z)) - (r_{i-1}(x) + r_{i-1}(y)) \\ &< r_i(y) + r_i(z) - 2\, r_{i-1}(x) < r_i(x) + r_i(z) - 2\, r_{i-1}(x)\end{aligned}$$

Análise amortizada dos splay steps

Caso do rr splay step.

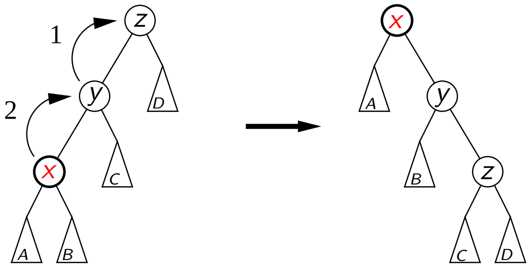


Custo real: 2

Alteração no potencial: $\Phi_i - \Phi_{i-1} < r_i(x) + r_i(z) - 2r_{i-1}(x)$

Análise amortizada dos splay steps

Caso do rr splay step.



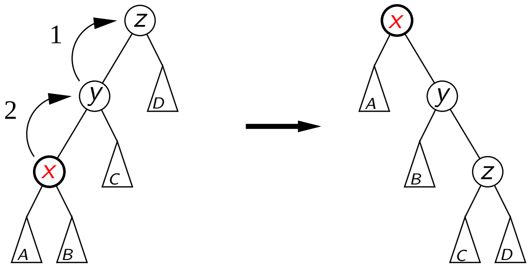
Custo real: 2

Alteração no potencial: $\Phi_i - \Phi_{i-1} < r_i(x) + r_i(z) - 2r_{i-1}(x)$

Custo amortizado: $\hat{c}_i \;<\; 2 + r_i(x) + r_i(z) - 2r_{i-1}(x)$

Análise amortizada dos splay steps

Caso do rr splay step.



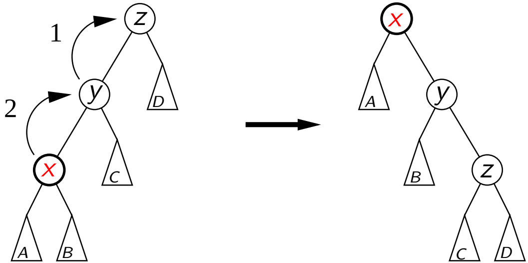
Custo real: 2

Alteração no potencial: $\Phi_i - \Phi_{i-1} < r_i(x) + r_i(z) - 2r_{i-1}(x)$

Custo amortizado: $\hat{c}_i \; < \; 2 + r_i(x) + r_i(z) - 2r_{i-1}(x)$

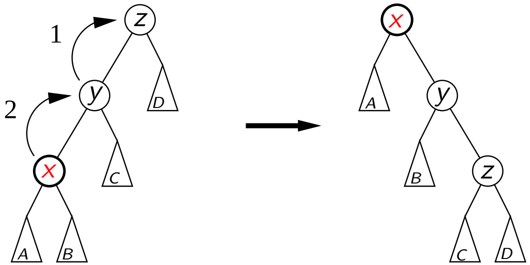
Queremos uma delimitação que dependa apenas de x.

Caso do rr splay step



Temos que $\frac{\log a + \log b}{2} \leq \log(\frac{a+b}{2})$ (função log é côncava). Logo

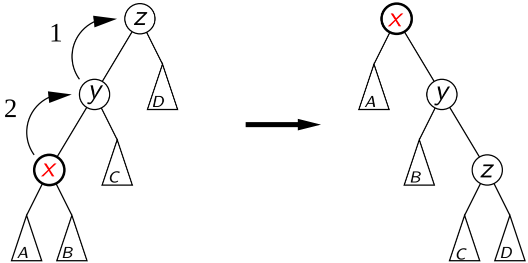
Caso do rr splay step



Temos que $\frac{\log a + \log b}{2} \leq \log(\frac{a+b}{2})$ (função log é côncava). Logo

$$\begin{aligned} r_{i-1}(x) + r_i(z) &= \lg s_{i-1}(x) + \lg s_i(z) \\ &\leq 2 \lg \Big(\frac{s_{i-1}(x) + s_i(z)}{2}\Big) \\ &< 2 \lg \Big(\frac{s_i(x)}{2}\Big). \end{aligned}$$

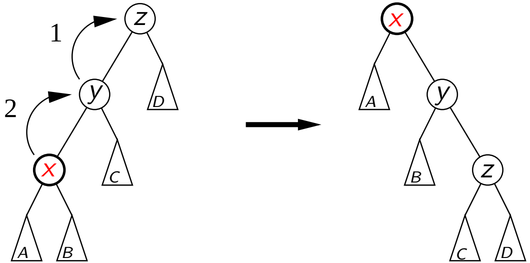
Caso do rr splay step



Temos que $\frac{\log a + \log b}{2} \leq \log(\frac{a+b}{2})$ (função log é côncava). Logo

$$r_{i-1}(x) + r_i(z) \;<\; 2 \lg s_i(x) - 2 \;=\; 2\, r_i(x) - 2.$$

Caso do rr splay step



Temos que $\frac{\log a + \log b}{2} \leq \log(\frac{a+b}{2})$ (função log é côncava). Logo

$$r_{i-1}(x) + r_i(z) \ < \ 2 \lg s_i(x) - 2 \ = \ 2\, r_i(x) - 2.$$

Então

$$\begin{aligned}
\hat{c}_i &< 2 + r_i(x) + r_i(z) - 2r_{i-1}(x) \\
&< 2 + r_i(x) + (2\, r_i(x) - 2 - r_{i-1}(x)) - 2r_{i-1}(x) \\
&= 3\,(r_i(x) - r_{i-1}(x)).
\end{aligned}$$

Análise amortizada dos splay steps

Analogamente podemos mostrar que

$$\hat{c}_i < 3\,(r_i(x) - r_{i-1}(x))$$

para rl splay steps, lr splay steps, e ll splay steps.

Análise amortizada dos splay steps

Analogamente podemos mostrar que

$$\hat{c}_i < 3\,(r_i(x) - r_{i-1}(x))$$

para rl splay steps, lr splay steps, e ll splay steps.

Para l splay steps e r splay steps, vale que

$$\hat{c}_i \leq 3\,(r_i(x) - r_{i-1}(x)) + 1.$$

Análise amortizada dos splay steps

Analogamente podemos mostrar que

$$\hat{c}_i < 3\,(r_i(x) - r_{i-1}(x))$$

para rl splay steps, lr splay steps, e ll splay steps.

Para l splay steps e r splay steps, vale que

$$\hat{c}_i \leq 3\,(r_i(x) - r_{i-1}(x)) + 1.$$

Se m é o número de splay steps e n o número de nós,

$$\begin{aligned} \sum_{i=1}^{m} \hat{c}_i &\leq 3\sum_{i=1}^{m}(r_i(x) - r_{i-1}(x)) + 1 \\ &= 3(r_m(x) - r_0(x)) + 1 \\ &\leq 3\lg n + 1. \end{aligned}$$