

Bacharelado em Ciência da Computação – UFU
Disciplina GBC053 – Gerência de Banco de Dados
Profa. Sandra de Amo

Solução dos Exercícios – Aula 28
Cálculo de Custos de Planos de Execução de Consultas

R(sid,bid,day,rname) : RESERVA

S(sid,sname,rating,age) : SAILORS

B(bid,bname, color) : BOAT

Consulta: Dê todos os nomes de marinheiros com rating > 5 que reservaram o barco com bid = 100.

Dados a considerar para o cálculo de custos dos planos:

Nº de páginas de R = M = 1000, com 100 tuplas por página

Nº de páginas de S = N = 500, com 80 tuplas por página

Nº de barcos reservados = 100 (existem 100 valores de bids distintos na tabela R)

Nº de valores para o atributo rating em S: 10 (varia de 1 a 10)

Hipóteses consideradas para a estimativa dos custos :

- As reservas estão uniformemente distribuídas entre os barcos.
- O número de marinheiros para cada valor de rating é aproximadamente o mesmo (distribuição uniforme dos valores de rating entre os marinheiros)

Nº de páginas livres no buffer = 5

Exercício 1:

$\text{Custo} = M + M.N = 1000 + 1000 \cdot 500 = \mathbf{501.000 \text{ I/Os}}$
--

Exercício 2:

Tamanho de Temp 1 = $1000/100 = 10$

Tamanho de Temp 2 = $500/2 = 250$

Custo de ordenação de Temp1 = $2 \cdot 10 \cdot n^\circ \text{ de etapas}$

Nº de etapas = $\lceil \log_4 10/5 \rceil + 1 = \lceil \log_4 2 \rceil + 1 = \lceil 0,5 \rceil + 1 = 2$

Custo de ordenação de Temp 1 = 40

Custo de ordenação de Temp2 = $2 \cdot 250 \cdot n^\circ \text{ de etapas}$

Nº de etapas = $\lceil \log_4 250/5 \rceil + 1 = \lceil \log_4 50 \rceil + 1 = \lceil 2,5 \rceil + 1 = 4$

Custo de ordenação de Temp 2 = 2000

Custo do Sort-Merge = $10 + 250 = 260$

Custo total do plano = $(1000+10+500+250) + (40 + 2000 + 260) = 1760 + 2300 =$
4060 I/O

Exercicio 3

$B-2 = 3$

Nº de blocos de 3 páginas = $\lceil 10/3 \rceil = 4$

Custo da operação de junção de Temp1 com Temp2 = $4*250 + 10 = 1010$

Custo total do plano de execução = $(1000 + 10 + 500 + 250) + 1010 =$ **2770 I/O**

Exercicio 4

Reserva tem 4 atributos. Sailors tem 4 atributos.

Tamanho de Temp1 após a projeção = $\lceil 10/4 \rceil = 3$ páginas

Tamanho de Temp2 após a projeção = $\lceil 250/2 \rceil = 125$ páginas

Nº de blocos de 3 páginas para Temp1 = **1**

Custo do Block Nested = $3 + 1 * 125 = 128$

Custo de criação e armazenamento das relações temporárias projetadas = $1000 + 3 + 500 + 125 = 1628$

Custo total do plano de execução = $1628 + 128 =$ **1756 I/Os**

Exercicio 5

- Custo para obter a relação T1 do lado esquerdo do Join
Observação: T1 não será armazenada, o Join será executado em pipeline a medida que as tuplas de T1 forem sendo criadas.

R tem 100.000 tuplas

Total de tuplas com bid 100 é = $100.000/100 = 1000$

Total de páginas para as tuplas com bid 100 = 10

Supondo que o índice hash segue a alternativa 2

Custo para encontrar a página do índice = 1,2

Como o índice é agrupado, as tuplas de dados com bid 100 estão agrupadas de acordo com o agrupamento do índice. Portanto, o custo para encontrar as tuplas de dados com bid 100 é = 10 (número de páginas contendo tais tuplas – cada página terá de ser acessada uma única vez).

Logo, custo de obter a relação T1 = $10 + 1,2 = 11,2$

- Custo do Join
Para cada tupla de T1 (1000 tuplas ao todo) temos uma única tupla correspondente em S.
- Caso o índice hash em Sailors utilize a alternativa (2)
 - Custo para encontrar esta única tupla = $1,2 + 1 = 2,2$
 - Custo total do Join = $1000 \cdot 2,2 = 2200$

Custo total do plano de execução = $11,2 + 2200 = 2211,2$ I/Os
--

- Caso o índice hash em Sailors utilize a alternativa (1)
 - Custo para encontrar esta única tupla = $1,2 = 1,2$
 - Custo total do Join = $1000 \cdot 1,2 = 1200$

Custo total do plano de execução = $11,2 + 1200 = 1211,2$ I/Os
--

Observações:

- 1) Como Sid é chave de Sailors, não faz diferença (em termos de custos) se o índice hash de Sailors é agrupado ou não.
- 2) Note que neste caso NÃO VALE A PENA “EMPURRAR” a seleção sobre rating > 5 para baixo do Join. Pelos seguintes motivos:
 - A seleção envolveria uma varredura em Sailors, já que o índice hash de que dispomos é no atributo SID e não no atributo *rating*.
 - Mesmo se tivéssemos um índice em rating para facilitar a seleção em Sailors, o índice em SID seria inútil para realizar o Join depois.

Conclusão: Não vale a pena “empurrar” a seleção para dentro do Join quando existem índices úteis para o Join que seriam inutilizáveis caso a seleção fosse “empurrada” para dentro.

Exercício 6

- Custo para criar e gravar (materializar) a relação temporária Temp1 ORDENADA (por SID) do lado esquerdo do Join:

$$1,2 + 10 + 10 + 2 \cdot 2 \cdot 10 = 61,2$$

Todas as tuplas com o mesmo SID em Temp1 estão juntas após a ordenação

- Temos 40.000 tuplas em Sailors
1000 reservas de barcos com bid 100
 - Suponhamos que **somente 4 marinheiros (M1, M2, M3, M4)** fizeram reservas de barcos.

(se todos os marinheiros tivessem reservado todos os barcos, teríamos pelo menos 4 milhões de reservas, o que não é verdade, pois temos 100.000 tuplas em Reservas !!)

- Suponhamos que cada um dos 4 marinheiros reservou todos os barcos o mesmo número de vezes.
- Assim, temos 250 reservas de cada barco por cada marinheiro.

Logo, para cada página de Temp1, temos 25 tuplas que combinam com o **M1**, 25 com o marinheiro **M2**, 25 com o **M3** e 25 com o **M4**.

Temp1 está ordenada por SID.

Logo, cada página de Temp1 é organizada da seguinte maneira:

100	M1	
100	M1	
100	M2	
100	M2	
100	M3	
100	M3	
100	M4	
100	M4	

Cada uma das faixas coloridas comporta 25 tuplas

Para cada página de Temp1, preciso buscar 4 páginas de Sailors no máximo (no exemplo, as páginas onde está o Sid = M1, o Sid = M2, o Sid = M3 e o Sid = M4). O custo para encontrar cada uma destas páginas com o índice hash é $1,2 + 1 = 2,2$.

Logo, o custo do Join é: $10 * 4 * 2,2 = 88$

Custo total do plano de execução = $61,2 + 88 = 148,2$ I/O
--

Obs. Quanto menor for o número de marinheiros que reservaram barcos, menor o custo desta consulta.

Exercício 7

Lado esquerdo do Join, supondo a alternativa 1 para o índice em Reserva: usando o índice, o custo da primeira seleção é **$10 + 1,2$ ($1,2 =$ custo para encontrar a 1ª página com bid=100, $10 =$ custo para encontrar o restante das páginas com bid=100 – já que o índice é agrupado).**

O resultado da seleção on the fly sobre o atributo Day fornece uma única tupla **t**, já que (bid, day) é chave de Reserva. Só existe uma única tupla em Sailors que combina com

esta tupla t . Para encontrá-la, usando o índice hash em Sailors (supondo a alternativa 1 para este índice) temos um custo de 1,2.

Logo, o custo total do plano de execução é $11,2 + 1,2 = \mathbf{12,4}$ I/Os
--

Observação: Este é um exemplo que mostra que empurrar uma seleção para dentro do Join (no caso a seleção sobre os atributos Day e Bid) é muito vantajosa, no caso dos atributos constituírem uma chave da relação selecionada (no caso Reserva).