

d) (artefato do tipo texto) Escolha 1 consulta dentre as quatro definidas na Parte I do trabalho. Reescreva sua consulta usando duas estratégias equivalentes (veja a teoria estudada em sala de aula), portanto criando 2 novas consultas, e execute novamente as análises de plano e custo. Verifique se o SGBD foi capaz de chegar no mesmo plano de consulta para execução de cada consulta equivalente. Analise o comportamento obtido (os mesmos itens de relatórios e prints do item (b) devem ser expostos aqui para fins de comparação). O texto deve ser entregue em formato PDF. O código deverá ser entregue em formato ASCII (.txt) na plataforma adotada pelo professor da sua turma (e-disciplina) (ver observação em vermelho abaixo).  
**(os mesmos itens de relatórios e prints do item (b) fazer interpretação e relação com teoria**

#### ORIGINAL:

```
"QUERY PLAN"
"Sort (cost=71.63..72.32 rows=278 width=44)"
"  Sort Key: produto.dataadquirida"
"    -> Hash Join (cost=31.84..60.34 rows=278 width=44)"
"      Hash Cond: ((SubPlan 1) = historico.idproduto)"
"        -> Seq Scan on produto (cost=0.00..17.40 rows=740 width=9)"
"        -> Hash (cost=30.90..30.90 rows=75 width=40)"
"          -> Hash Join (cost=2.69..30.90 rows=75 width=40)"
"            Hash Cond: (cliente.idcliente = historico.idcliente)"
"              -> Seq Scan on cliente (cost=0.00..22.70 rows=1270 width=36)"
"              -> Hash (cost=1.75..1.75 rows=75 width=8)"
"                -> Seq Scan on historico (cost=0.00..1.75 rows=75 width=8)"
"      SubPlan 1"
"        -> Result (cost=0.01..0.01 rows=1 width=4)"
"          One-Time Filter: (produto.vendido AND (produto.dataadquirida >=
'2017-01-01'::date) AND (produto.dataadquirida <= '2020-12-31'::date))"
```

#### NOVA CONSULTA:

```
SELECT cliente.nome, cliente.idcliente, produto.idproduto
FROM historico
INNER JOIN produto ON historico.idproduto = produto.idproduto
INNER JOIN cliente ON historico.idcliente = cliente.idcliente
WHERE (produto.dataadquirida BETWEEN '2017-01-01' AND '2020-12-31') AND
(produto.vendido = true)
ORDER BY dataadquirida;
```

```
"QUERY PLAN"
"Sort (cost=23.89..23.90 rows=1 width=44)"
"  Sort Key: produto.dataadquirida"
"    -> Nested Loop (cost=21.28..23.88 rows=1 width=44)"
"      -> Hash Join (cost=21.13..23.07 rows=1 width=12)"
"        Hash Cond: (historico.idproduto = produto.idproduto)"
"          -> Seq Scan on historico (cost=0.00..1.75 rows=75 width=8)"
```

```

“      -> Hash (cost=21.10..21.10 rows=2 width=8)”
“      -> Seq Scan on produto (cost=0.00..21.10 rows=2 width=8)”
“      Filter: (vendido AND (dataadquirida >= '2017-01-01'::date) AND
(dataadquirida <= '2020-12-31'::date))”
“      -> Index Scan using id_cliente_id on cliente (cost=0.15..0.81 rows=1 width=36)”
“      Index Cond: (idcliente = historico.idcliente)”

```

### **Interpretação:**

Index Scan using id\_cliente\_id on cliente (cost=0.15..0.81 rows=1 width=36)

Leitura das tabelas performada em uma estrutura de árvore B, onde ela percorre até as folhas para conseguir alcançar os registros relevantes para a relação que respeite a condição (Index Cond: (idcliente = historico.idcliente)).

Seq Scan on produto (cost=0.00..21.10 rows=2 width=8)

Leitura sequencial da tabela produto.

Hash (cost=21.10..21.10 rows=2 width=8)

Construção e armazenamento dos registros em uma tabela hash.

Seq Scan on historico (cost=0.00..1.75 rows=75 width=8)

Leitura sequencial da tabela historico.

Hash Join (cost=21.13..23.07 rows=1 width=12)

Leitura da tabela histórico de um registro cada vez para serem colocados na memória em uma estrutura hash para que depois, ao ler a segunda tabela de produto, é comparado a correspondência de cada registro com relação às da primeira tabela respeitando a condição (Hash Cond: (historico.idproduto = produto.idproduto)).

Nested Loop (cost=21.28..23.88 rows=1 width=44)

Leitura de um outer table, e para cada registro do outer table que respeite a condição, há leitura de um registro de uma inner table.

Sort (cost=23.89..23.90 rows=1 width=44)

Organização física da tabela.

### **Relação:**

As principais diferenças entre os planos foi a utilização de um Index Scan da nova consulta com relação ao Hash, além do custo ter sido bem menor, já que para as consultas, o SGBD apenas lidou com as tabelas das relações intermediárias reduzidas devido a antecipação das projeções.