Atividade Prática 01

COM231 – Banco de Dados 2

- Tabela possui 18.674 registros (r)  
- Cada bloco possui 8kb – 8.192 bytes  
- O ponteiro de índice ocupa 4 bytes

**1 - a)** PostgreSQL 🡪 tabela heap

INTEGER = 4 bytes Ponteiro = 4 bytes

8 bytes por registro de índice 🡪 8.192/ 8 = 1024 registros por bloco

Método 1 – número de registros / número de registros por bloco

N\_blocos = 18.674 / 1024 = 18,23 = 19 blocos

Método 2 – (nro de registros \* (tamanho do índice + tamanho do ponteiro)) / tamanho do bloco

T = 18.674 \* (4 + 4) = 18.674 \* 8 = 149.392 bytes

N\_blocos = 149.392 / 8.192 = 18,23 = 19 blocos

**O índice ocupa 19 blocos da memória.**

**O índice é classificado como denso quanto a quantidade de entradas, porque existe uma entrada de índice para cada registro da tabela, devido a organização heap de registros de arquivo utilizado pelo PostgreSQL.**

**O índice é classificado como primário quanto a organização do arquivo, porque é utilizada uma chave primária simples (única e sem chave composta).**

**1 - b)**

Tamanho do registro = 4 + 35 = 39 bytes🡪 8.192/ 39 = 210 registros por bloco

N\_blocos = 18.674 / 210 = 88,9 = 89 blocos

**O índice ocupa 89 blocos da memória.**

**O índice é classificado como denso quanto a quantidade de entradas por existir uma entrada de índice para cada registro da tabela, devido organização heap de registros de arquivo utilizado pelo PostgreSQL.**

**O índice é classificado como secundário pois utiliza como índice um atributo que não é nem a chave primária e nem composto em conjunto com a chave primária (índices secundários são sempre densos).**

**2 –** Por utilizar arquivos sequenciais o índice primário sempre será esparso e o secundário será denso.

**a)** 8 bytes por registro de índice

4 + 35 + 4 + 4 + 20 + 2 + 2 + 2 + 1 + 17 = 91 bytes p registro = 90 registros por bloco = 208 blocos

8 bytes para cada bloco 🡪 8 \* 208 = 1664 bytes = 1 bloco

**O índice ocupa 1 bloco da memória.**

**O índice é classificado como esparso quanto a quantidade de entradas, porque existe uma entrada de índice para um conjunto de registros da tabela, devido a utilização de arquivos sequenciais o índice primário sempre será esparso.**

**O índice é classificado como primário quanto a organização do arquivo, porque é utilizada uma chave primária simples (única e sem chave composta).**

**b)**

Tamanho do registro = 4 + 35 = 39 bytes 🡪 210 registros por bloco ()

N\_blocos = 18.674 / 210 = 88,9 = 89 blocos

**O índice ocupa 89 blocos da memória.**

**O índice é classificado como denso quanto a quantidade de entradas, porque existe uma entrada de índice para cada conjunto de registros da tabela, a utilização de arquivo sequencial nesse caso não importa porque índices secundários são sempre densos.**

**O índice é classificado como secundário pois utiliza como índice um atributo que não é nem a chave primária e nem composto em conjunto com a chave primária).**