Gerenciamento de disco e buffer

Introdução

SGBD armazena dados no disco

Isso é que mais implica no projeto de um SGBD

- Operações de escrita (write);
- Operações de leitura (read)

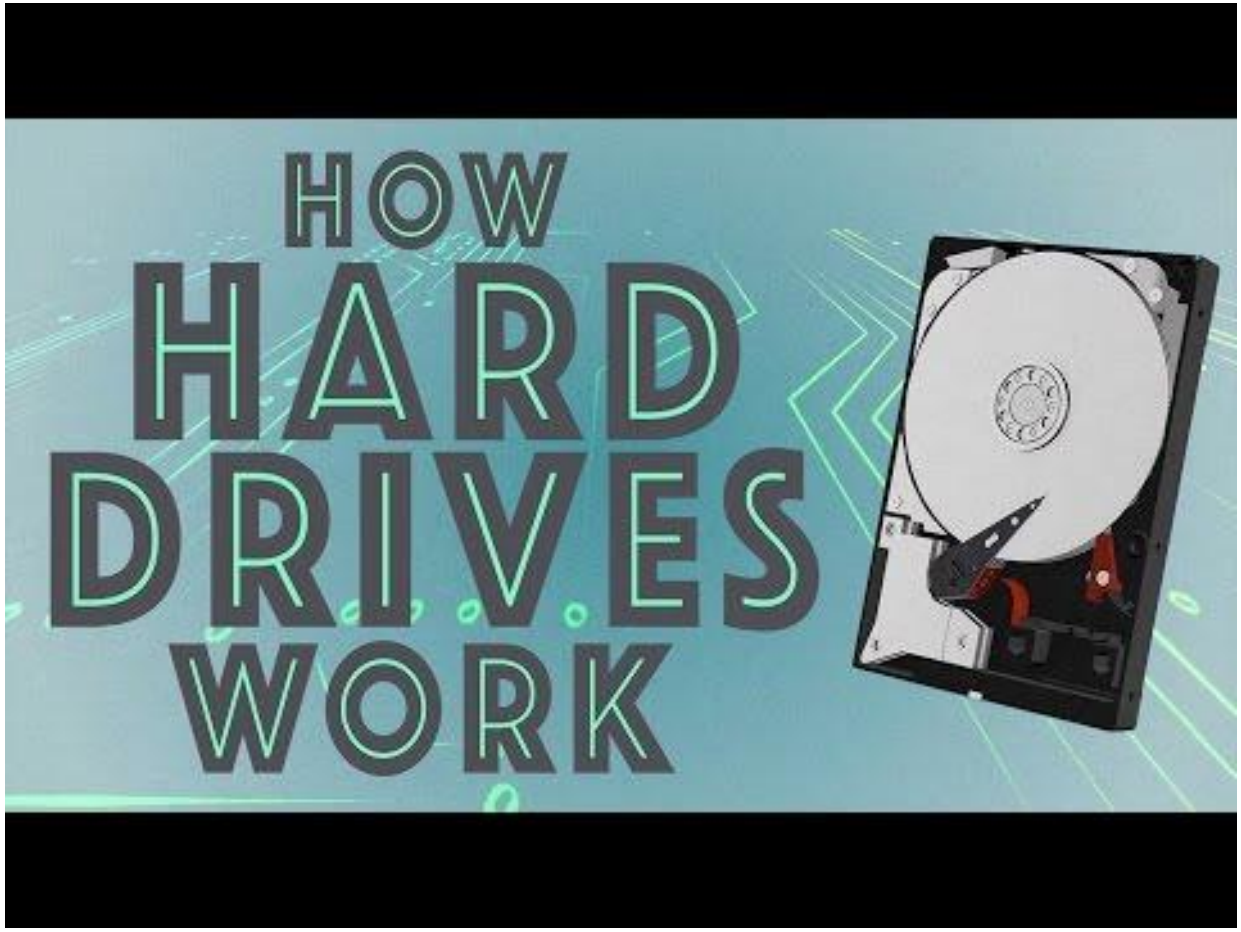
Ambas as operações têm um custo muito alto (tempo) e devem ser planejadas cuidadosamente

Discos

Dados são armazenados e recuperados em unidades chamadas *blocos de disco*.

O tempo para recuperar um bloco no disco depende da posição onde se encontra.

Componentes Disco



SSD vs HD

Acessar conteúdo do link:

Atividade 1

1- Explique o motivo de uma escrita de um bloco no SSD ser mais lenta que uma leitura.

2- Você acredita que o disco rígido será descontinuado nos próximos anos? Explique.

3- Explique como acontece uma operação de atualização no SSD vs HD.

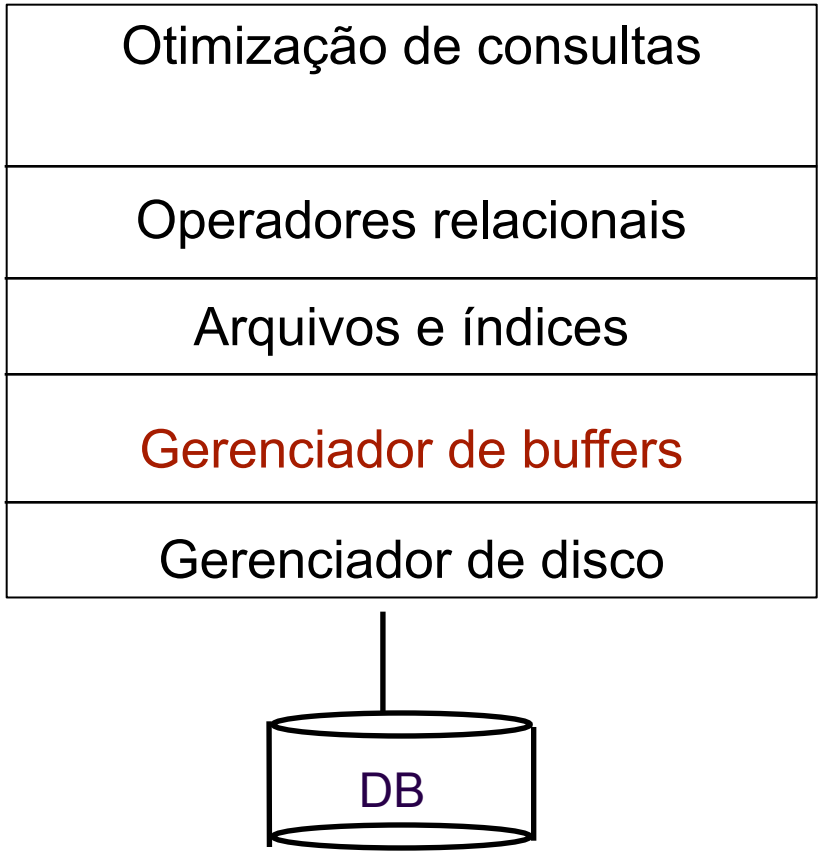
4- Até então vimos na disciplina o sistema transacional, log e concorrência. Qual (ou quais) desses mecanismos é (são) mais impactado(s) com a utilização de um disco de alta velocidade SSD? Explique

Atividade 2

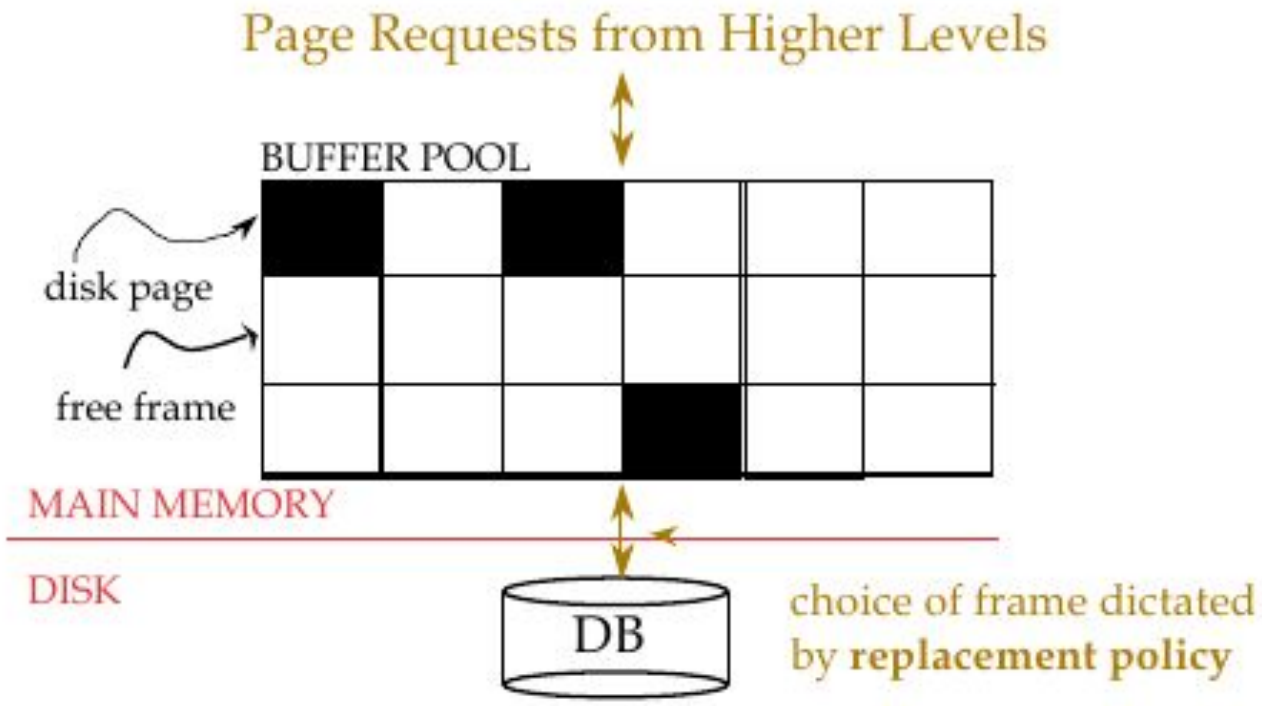
Árvores B+ são bastante utilizadas na indexação no SBGD devido sua capacidade de processar milhões de chaves com um número reduzido de níveis e tempo constante de acesso. O objetivo desta atividade é que seja revisado o conteúdo de B+ a partir da execução de inserções.

A) Insira em uma árvore B+ com nível 3(cada nó armazena 2 chaves) as seguintes chaves: 30 20 15 17 18 5 9 8 7 6 5 4 3 e 2 1. Deve ser explicado cada operação no processo de inserção.

Banco de dados relacional



Gerenciador de buffers



[Ramakrishna et. al. 3 edição, pag 264]

Otimização do Acesso

Buffer: porção da memória principal que armazena cópias dos blocos do disco.

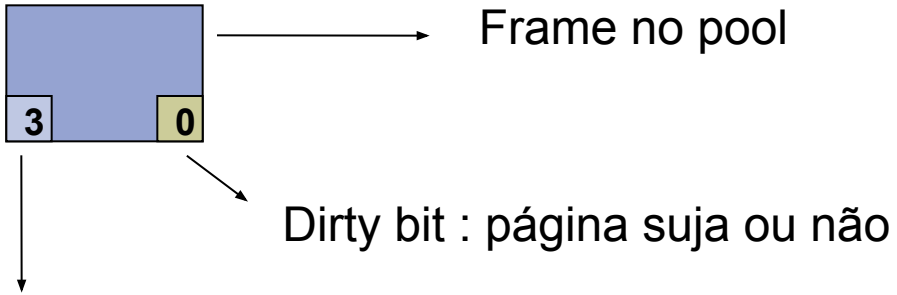
- O buffer é organizado em páginas que, geralmente, se relacionam 1:1 com os blocos

Gerenciador de buffer: subsistema de um SGBD responsável para gerenciar espaços no buffer

Gerenciador de Buffer

- Testa se o dado procurado está no buffer
- Traz a página do disco para a memória
- Procura frames livres (espaço memória) para alocar a página
- Aciona algoritmo para liberar a página
- Aloca página
- Caso o frame tiver que ser reutilizado, propaga modificação no disco.

Algoritmo de Gerenciamento



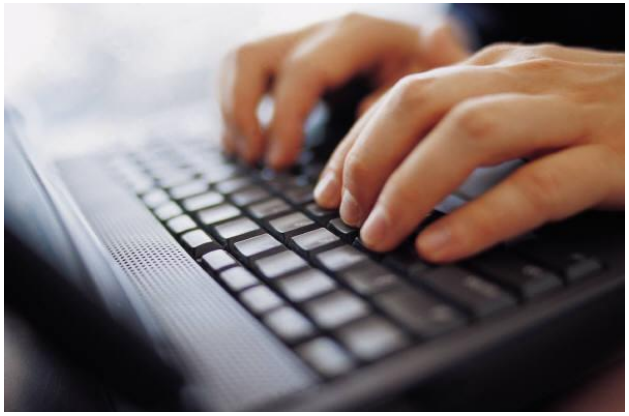
Pin-count = número de vezes que a página foi solicitada para consultas ou modificações mas não foi liberada ainda.

Inicialmente :

Dirty bit := 0

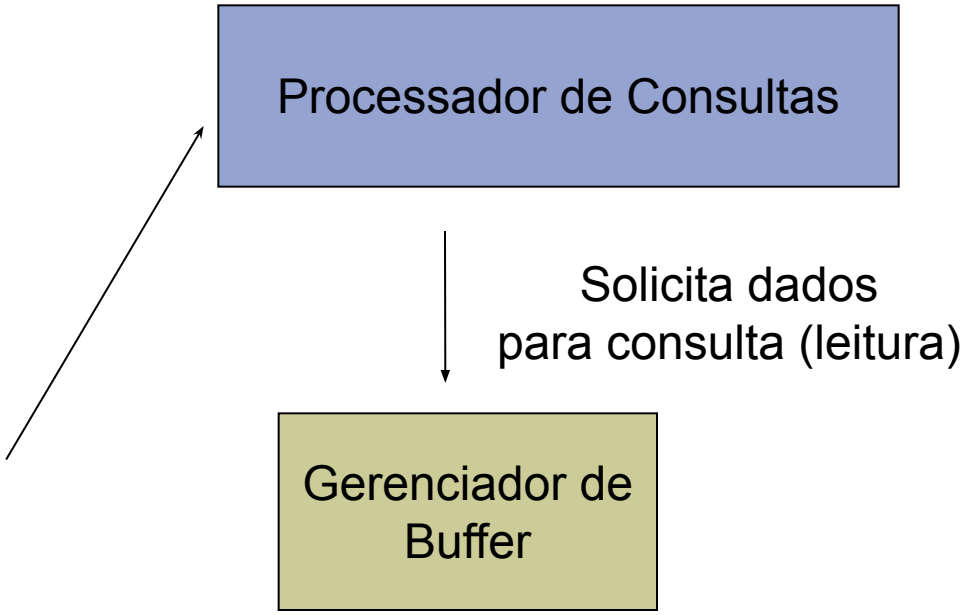
Pin-count := 0

Considere o seguinte cenário



Usuário consulta banco de dados

SELECT *
From EMP
WHERE EMP.CPF = 40333994598



Hipóteses

- Dados estão armazenados sequencialmente no disco na ordem em que foram inseridos

- Relação EMP está armazenada em um arquivo de nome EMP.

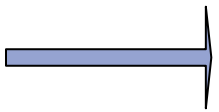
- Existe um arquivo no disco onde se armazena, para cada arquivo, o endereço no disco (cilindro, superfície, trilha, bloco) de sua primeira página.

  O buffer pool está com todos os frames ocupados!!!

14

O que faz o gerenciador de buffer (1)

- Verifica se a página 1 do arquivo EMP está no buffer pool e em que frame.
- **Caso positivo**:
  - informa ao processador de consultas o frame onde se encontra a página 1.
  - Incrementa *pinout* deste frame.
- **Caso negativo**: precisa encontrar espaço no buffer pool para alocar página 1



O que faz o gerenciador de buffer (2)

- Verifica se existem frames com *pinout* = 0
- **Caso positivo**: aciona gerenciador de disco
  - Gerenciador de disco posiciona cabeça de leitura sobre o endereço da primeira página do arquivo EMP (conhecido do gerenciador de buffer)
  - Gerenciador de disco providencia a transferência da página.
  - Gerenciador de buffer vai alocar a página em um frame com *pinout* = 0
  - Qual frame será escolhido ?
    - Usa sua política de substituição (LRU, MRU, random)
  - Verifica o dirty bit deste frame

O que faz o gerenciador de buffer (3)

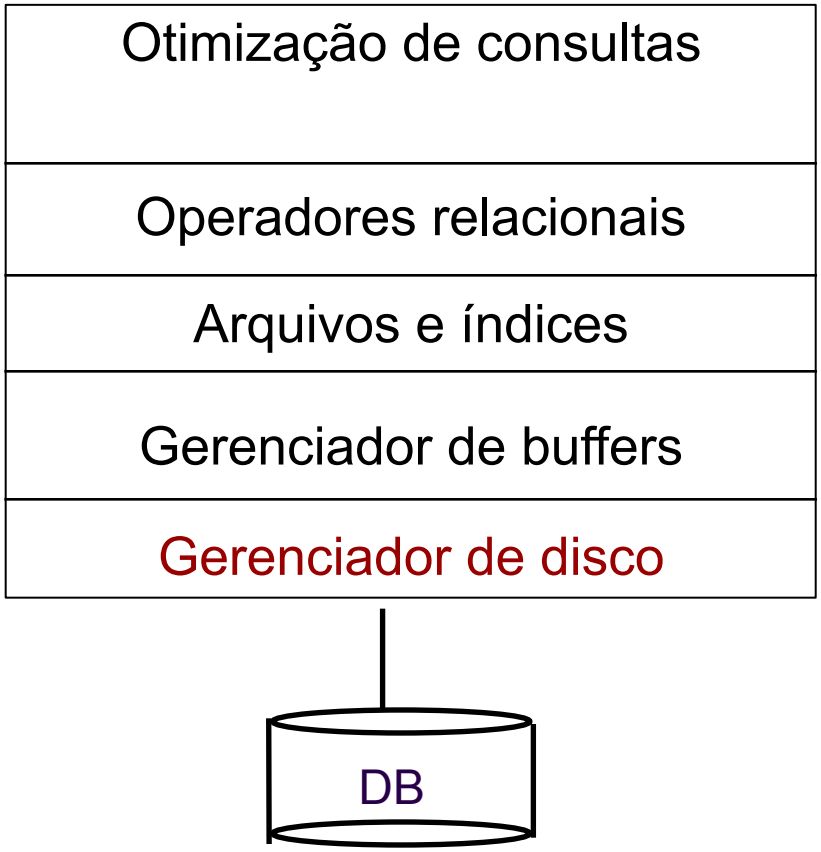
- **Dirty bit = 1**:
  - grava a página atual do frame no disco
  - Aloca a nova página no frame
  - Incrementa *pinout* do frame
  - Retorna o endereço do frame para o processador de consultas
- **Dirty bit = 0**:
  - Aloca a nova página no frame
  - Incrementa *pinout* do frame
  - Retorna o endereço do frame para o processador de consultas

O que faz o gerenciador de buffer (4)

- **Caso negativo**: não existe nenhum frame com *pinout* = 0
  - Gerenciador de buffer deve esperar até que um frame se libere (pinout = 0)
  - Caso o tempo de espera ultrapasse um certo limite, envia mensagem de erro para o processador de consultas.

18

Banco de dados relacional



Arquivos

Blocos são transmitidos entre disco e memória, mas...

O SGBD trabalha no nível de registro e arquivos

Arquivo: uma coleção de páginas, que contém um conjunto de registros. Estes devem suportar:

- Inserção/remoção/atualização
- Busca de registro em particular
- Busca de todos os registros

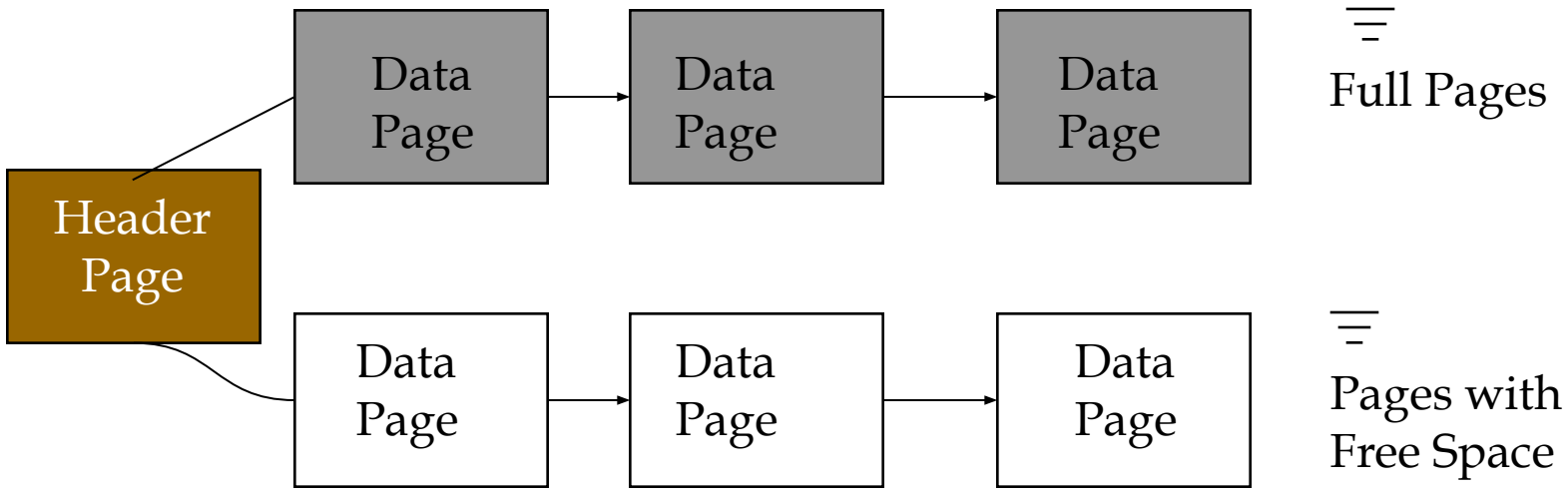
Arquivos não ordenados (Heap)

Estrutura mais simples de armazenar registros sem ordenamento

Para suportar as operações, é importante:

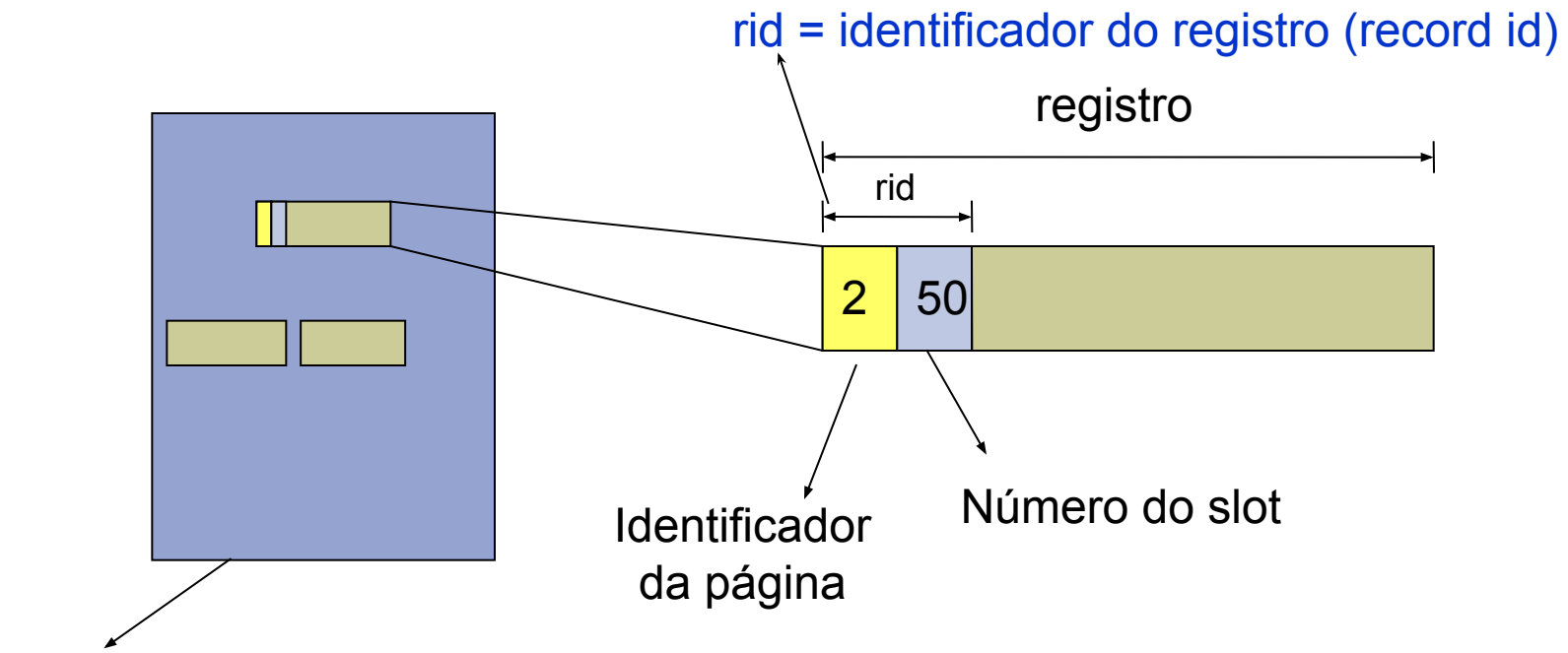
- Manter o rastro das páginas no arquivos
- Manter o rastro das páginas vazias
- Manter o rastro dos registros em cada página

Heaps como lista



O ponteiro da página inicial (header page) deve ser armazenado;

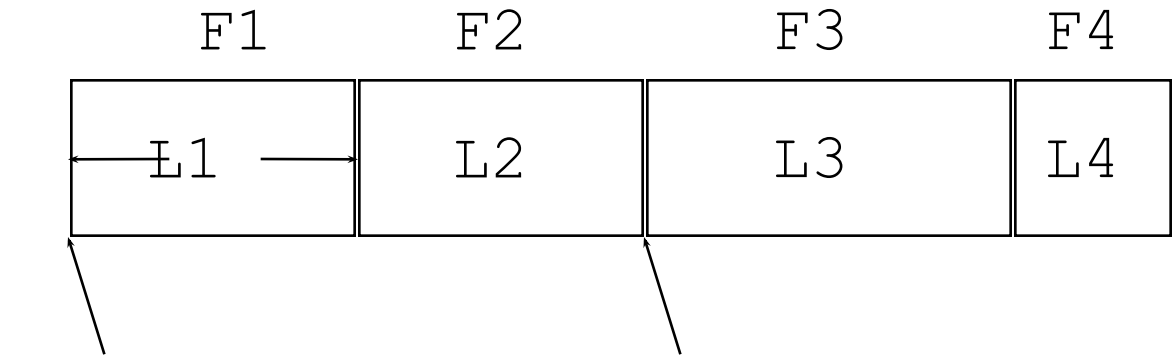
Como os registros são organizados nas páginas



Slot = espaço físico preenchido por um registro

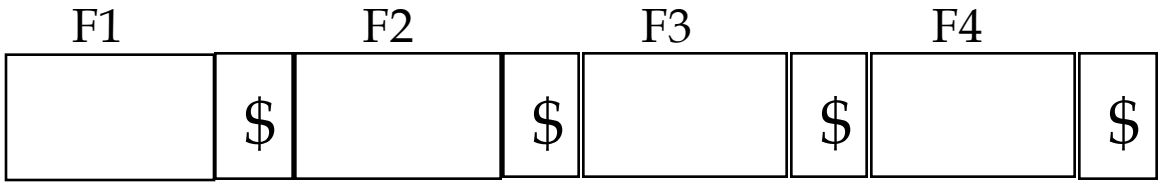
23

Formato fixo de registros:

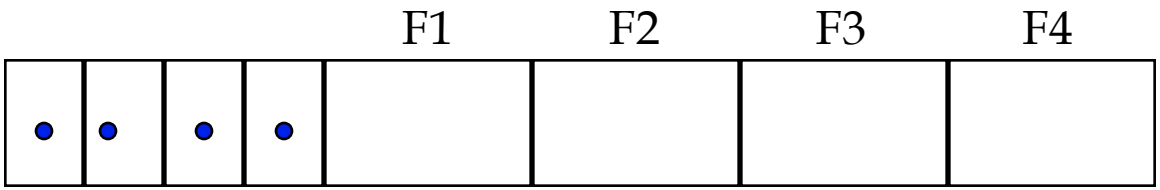


Registros com tamanho variável

Duas alternativas

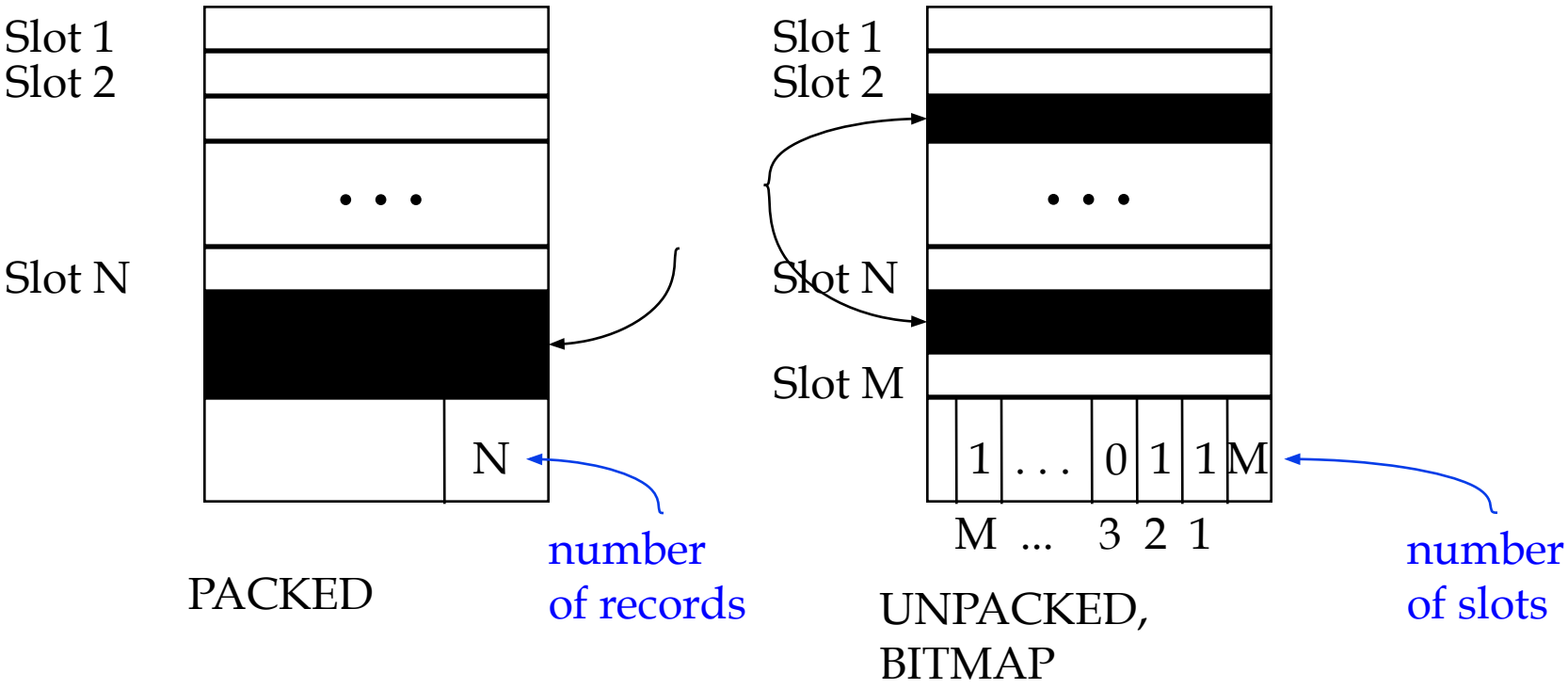


Campos determinados por caractere especial



Array de offsets

Páginas com formato fixo



- *Record id = <page id, slot #>*.

Exercício

create table Movie(name char(30), address char (255), data date*)

*date ocupa 10 bytes

Exemplo: Suponha que os registros da tabela Movie serão armazenados em páginas de 4kb. O cabeçalho do registro ocupa 12 bytes (ponteiro para o esquema, tamanho registro, timestamp). Quantos registros cabem na página?

Exercício

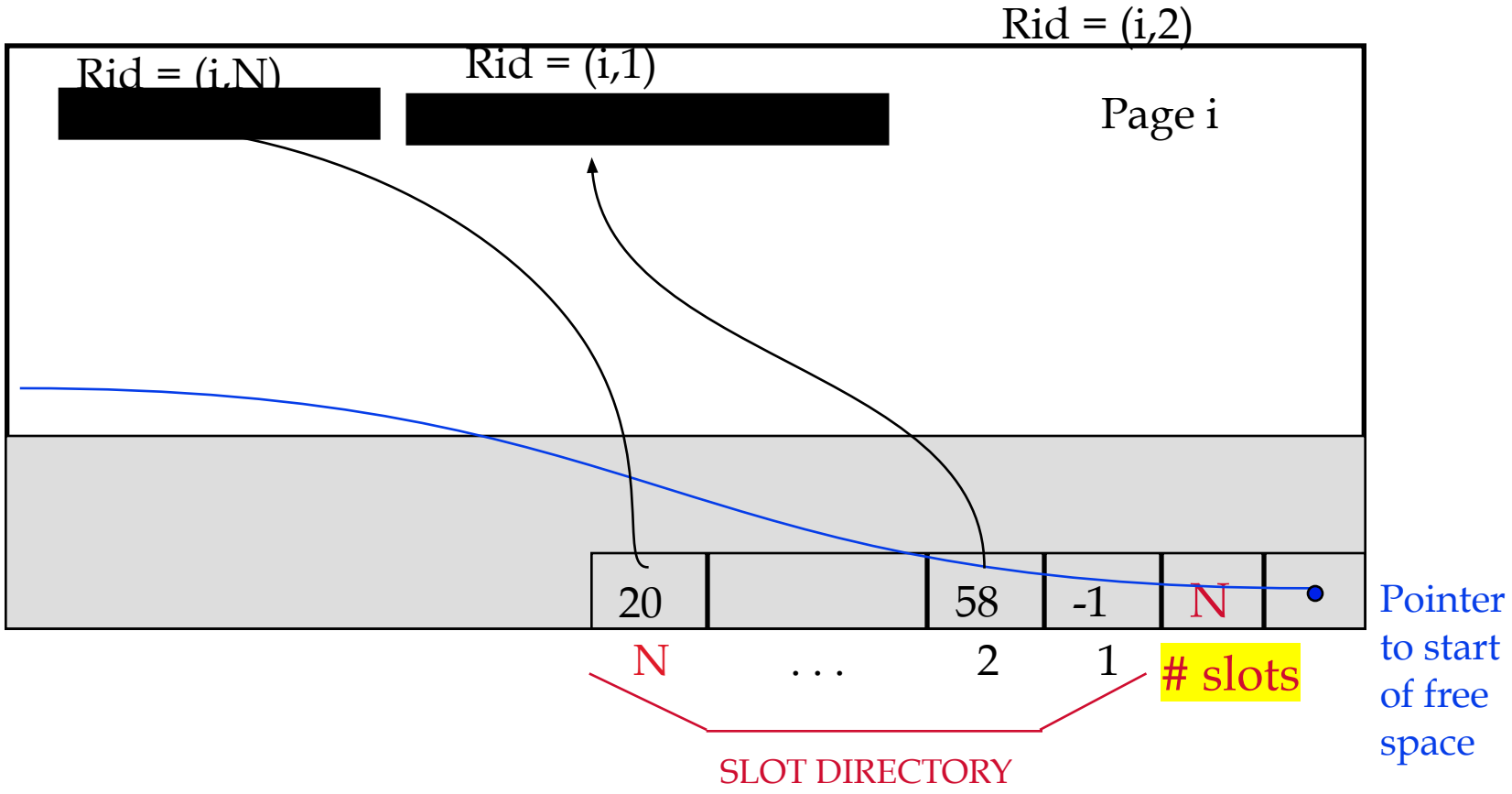
A)Construa um diagrama de página de tamanho fixo usando um mapa de bits. A tabela pode armazenar 20 registros de 12 bytes cada.

B) Insira 10 registros

C) Apague o registro 2 e 5 previamente inseridos

D) Insira 2 registros nos espaços anteriormente liberados

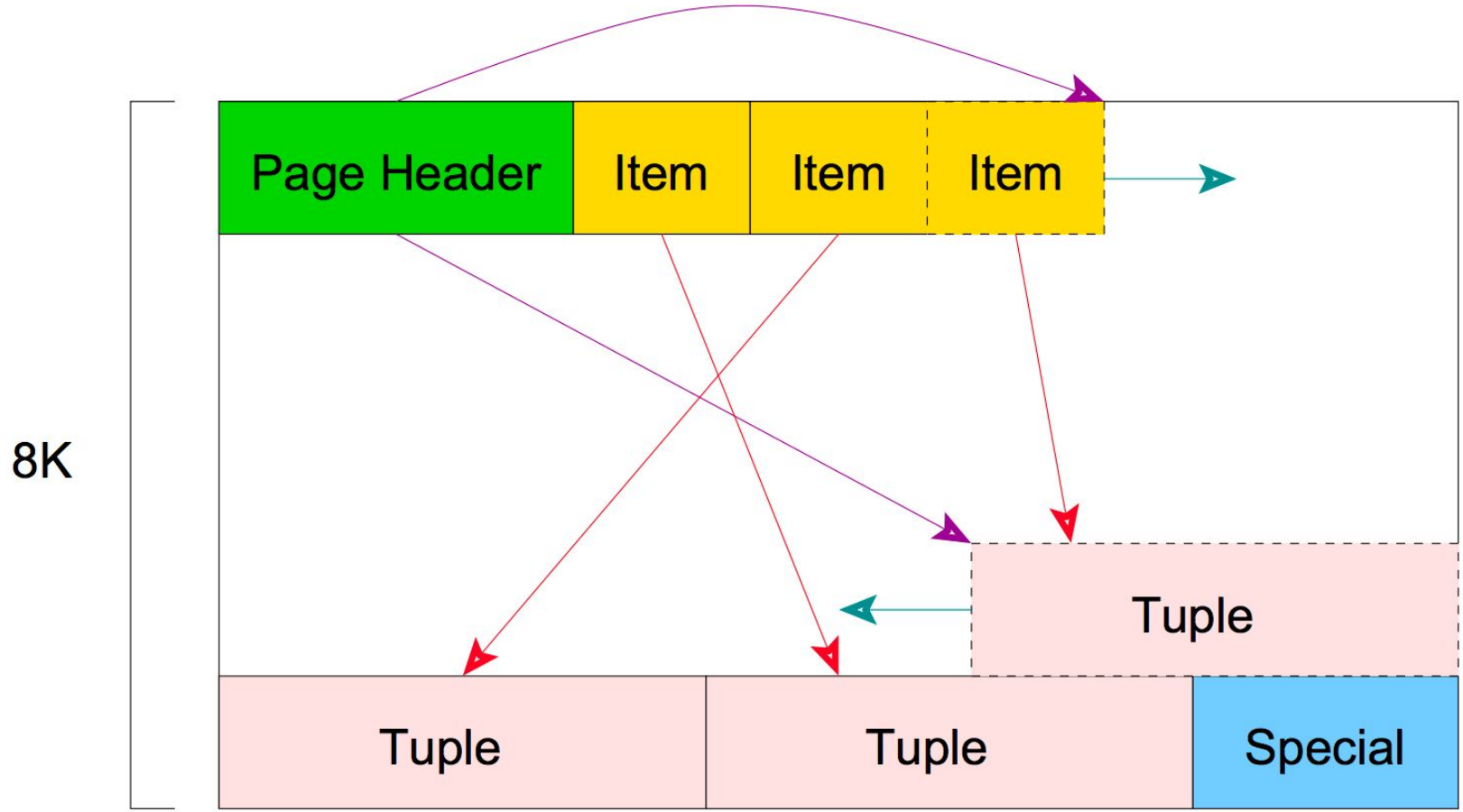
Registros com tamanhos variados



Exemplo

O que acontece no exemplo anterior, em relação ao espaço ocupado pelo registros, se o atributo “nome” fosse alterado para varchar(255)?

Postgres

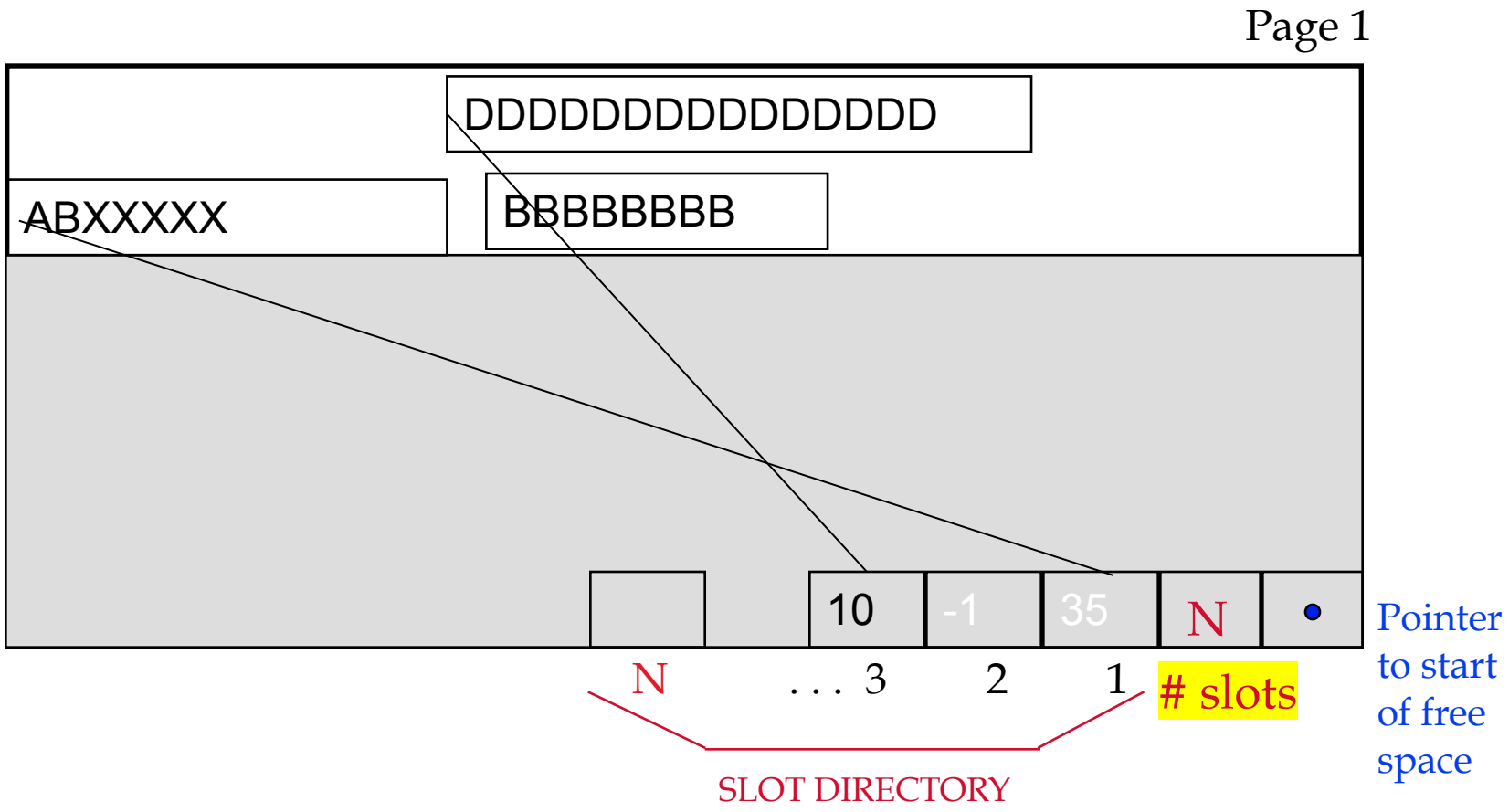


Atividade 1

Insira os registros abaixo em uma página com tamanho variado. Os registros tem 200 bytes no máximo cada. Tamanho total da página 4kb.

1) “A”
2) “BBBBBBBB”
3) “DDDDDDDDDDDDDDD"
4) O que acontece caso o primeiro registro seja alterado para “ABXXXXX”

Atividade 1



1- RID: 1 1
2- RID:
3- RID: 1 3

Atividade para entregar

Create table teste (varchar[30] x, int y)

A) Insira 3 registros na tabela acima.

B) Remova o primeiro dos registros anteriormente inseridos

C) Insira 1 novo registro.

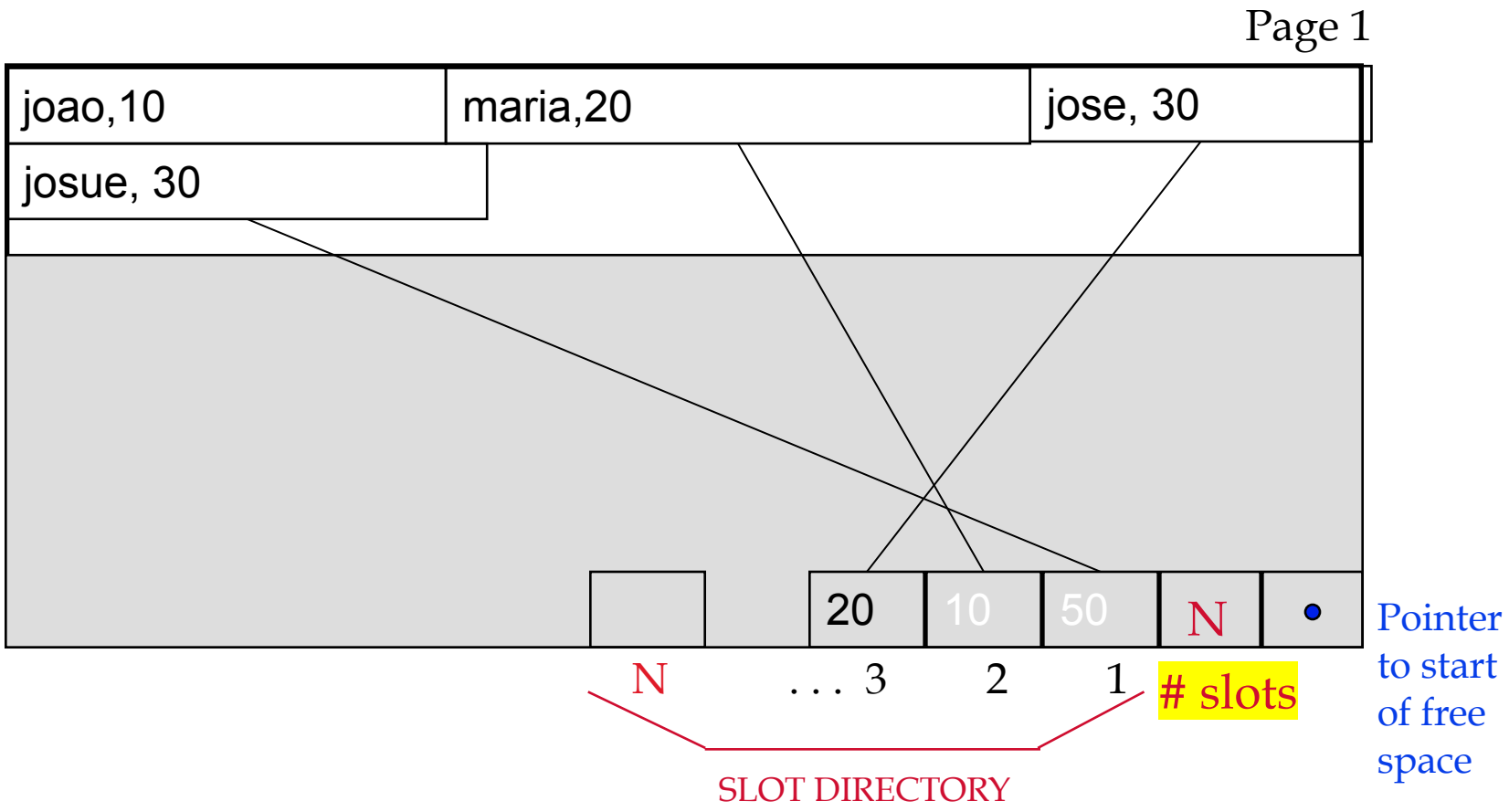
Create table teste (varchar[30] x, int y)

A) Insira 3 registros na tabela acima.

B) Remova o primeiro dos registros anteriormente inseridos

C) Insira 1 novo registro.

Atividade 1



1- RID:
2- RID: 1 2
3- RID: 1 3
4- RID: 1 1