

Campos e Registros

Adaptado dos Originais de:

Ricardo Campello
Thiago Pardo
Leandro C. Cintra
Maria Cristina F. de Oliveira

1

Organização de Arquivos

- Informações em arquivos são, em geral, organizadas logicamente em campos e registros
 - campos e registros são conceitos lógicos
- Dependendo de como a informação é mantida, campos lógicos sequer podem ser recuperados...

2

Seqüência de Bytes (stream)

- Exemplo:
 - Suponha que desejamos armazenar em um arquivo os nomes e endereços de várias pessoas
 - Suponha que decidimos representar os dados como uma seqüência simples de bytes
 - caracteres sem delimitadores, contadores, etc

3

Seqüência de Bytes (stream)

- Exemplo:

John Ames	Alan Mason
123 Maple	90 Eastgate
Stillwater, 74075	Ada, 74820

4

Seqüência de Bytes (stream)

- Exemplo:

John Ames	Alan Mason
123 Maple	90 Eastgate
Stillwater, 74075	Ada, 74820

AmesJohn123 MapleStillwater74075MasonAlan90 EastgateAda74820

5

Seqüência de Bytes (stream)

- Uma vez escritas as informações, em princípio não existe como recuperar as unidades lógicas (nome, endereço)
 - perde-se a integridade das unidades fundamentais de organização dos dados
 - essas unidades são agregados de caracteres
 - tais agregados de caracteres com significado
 - **campos** (fields)

6



- Formas de organização dos campos
- Formas de organização dos registros

7



Organização em Campos

- **Campo:**
 - **menor unidade lógica** de informação em arquivo
 - noção lógica (ferramenta conceitual)
 - não está associada a um conceito físico
 - Existem várias maneiras de organizar um arquivo mantendo a identidade dos campos
 - A organização anterior não proporciona isso...

8



Organização em Campos

- Estruturas de Organização de Campos:
 - Comprimento fixo
 - Indicador de comprimento
 - Delimitadores
 - Uso de *tags*

9



Campos com Tamanho Fixo

- Cada campo ocupa no arquivo um tamanho fixo, pré-estabelecido
 - por exemplo: 4 bytes
- O fato do tamanho ser conhecido garante que é possível recuperar cada campo

Maria	Rua 1	123	São Carlos
João	Rua A	255	Rio Claro
Pedro	Rua 10	56	Rib. Preto

11



Campos com Tamanho Fixo

- Exemplo em C:

```
struct set_of_fields {  
    char last_name[10];  
    char first_name[10];  
    char address[15];  
    char city[2];  
    char zip_code[9];  
};
```

12



Campos com Tamanho Fixo

- Desvantagens?

13

Campos com Tamanho Fixo

- O espaço alocado (e não usado) aumenta desnecessariamente o tamanho do arquivo
 - desperdício de memória secundária: **fragmentação**
- Solução inapropriada quando se tem uma grande variabilidade nos tamanhos dos campos
- Razoável apenas se o comprimento dos campos é realmente fixo, ou apresenta pouca variação

14

Campos com Indicador de Comprimento

- O tamanho de cada campo é armazenado imediatamente antes da informação
 - requer um único byte se o tamanho do campo é inferior a 256 bytes

15

Campos com Indicador de Comprimento

- O tamanho de cada campo é armazenado imediatamente antes da informação
 - requer um único byte se o tamanho do campo é inferior a 256 bytes

```
05Maria05Rua 10312310São Carlos
04João05Rua A0325509Rio Claro
05Pedro06Rua 10025610Rib. Preto
```

16

Campos Separados por Delimitadores

- Caractere especial (delimitador) inserido ao final de cada campo
 - delimitador não pode ser um caractere válido
 - espaços em branco não serviriam...
 - para ilustração, podemos utilizar "|", "#", ...

17

Campos Separados por Delimitadores

- Caractere especial (delimitador) inserido ao final de cada campo
 - delimitador não pode ser um caractere válido
 - espaços em branco não serviriam...
 - para ilustração, podemos utilizar "|", "#", ...

```
Maria|Rua 1|123|São Carlos|
João|Rua A|255|Rio Claro|
Pedro|Rua 10|56|Rib. Preto|
```

18

Campos com Tags "keyword=value"

- Vantagem:
 - possui semântica local
 - campo fornece informação sobre si próprio
 - permite identificar localmente o conteúdo do arquivo
 - um campo "perdido" não compromete o arquivo
 - permite campos existirem ou não
- Desvantagem:
 - as *keywords* podem ocupar uma porção significativa do arquivo

21

Campos com Tags "keyword=value"

- utilizado em combinação com outro método (delimitador)
 - nem sempre necessário, porém útil

```
Nome=Maria|Endereço=Rua 1|Número=123|Cidade=São Carlos|
Nome=João|Endereço=Rua A|Número=255|Cidade=Rio Claro|
Nome=Pedro|Endereço=Rua 10|Número=56|Cidade=Rib. Preto|
```

22

Organização em Registros

- **Registro:**
 - um conjunto de campos agrupados
- Arquivo organizado em registros
 - nível de organização mais alto
- Assim como os campos, um registro é uma ferramenta conceitual
 - está associado ao arquivo lógico
 - outro nível de organização imposto aos dados
 - Objeto de dado estruturado
 - Objeto (memória) <-> registro (arquivo)

23

Organização em Registros

- Estruturas de Organização de Registros:
 - Tamanho **fixo**
 - Campos de tamanho fixo
 - Campos de tamanho variável
 - Tamanho **variável**
 - Número pré-determinado de campos
 - Uso de delimitadores
 - Indicador de tamanho
 - Uso de índice

24

Registros de Tamanho Fixo

- Analogamente ao conceito de campos de tamanho fixo, assume que todos os registros têm o mesmo tamanho, com campos de tamanho fixo ou não
- Um dos métodos mais comuns de organização de arquivos

25

Registros de Tamanho Fixo

Registro de tamanho fixo e campos de tamanho fixo:

Maria	Rua 1	123	São Carlos
João	Rua A	255	Rio Claro
Pedro	Rua 10	56	Rib. Preto

Registro de tamanho fixo e campos de tamanho variável:

Maria	Rua 1	123	São Carlos	← Espaço vazio →
João	Rua A	255	Rio Claro	← Espaço vazio →
Pedro	Rua 10	56	Rib. Preto	← Espaço vazio →

26

Registros de Tamanho Fixo

- Assume que todos os registros ocupam o mesmo número de bytes:
 - campos podem ou não ocupar o mesmo no. de bytes
- Um dos métodos mais comuns de organização de arquivos:
 - simples,
 - Fácil atualização
 - **permite acesso direto aos registros**
- Porém, pode ser inapropriado...
 - desperdício de memória secundária – **fragmentação**

27

Registros de Tamanho Variável

- Ao invés de especificar que cada registro contém um número fixo de bytes, podemos especificar um **número fixo de campos**
 - O tamanho do registro é **variável em bytes**
 - Acesso direto inviabilizado
 - reconhecer campos contando e aplicando módulo nro-campos
 - Por ex., campos separados por delimitadores:

```
Maria|Rua 1|123|São Carlos|João|Rua A|255|Rio Claro|Pedro|Rua 10|56|Rib. Preto|
```

4 campos

4 campos

4 campos

28

29

Registros de Tamanho Variável

- Pode-se ainda separar **registros** com delimitadores
 - análogos aos de fim de campo
 - porém outro caractere deve ser utilizado
 - delimitador de campos pode ser mantido
 - Aplicativo deve saber qual o delimitador utilizado
 - Não há como calcular a posição do n-ésimo registro

Registros de Tamanho Variável

- Vantagens:
 - permite número variável de campos
 - início do arquivo não é mais a única referência
 - Sem fragmentação

```
Maria|Rua 1|123|São Carlos|#João|Rua A|255|Rio Claro|#Pedro|Rua 10|56|Rib. Preto|#
```

30

Registros de Tamanho Variável

- **Indicador de Tamanho:**
 - indicador que precede o registro
 - fornece o seu tamanho total, em bytes
 - No início de cada registro:
 - sabe-se onde termina aquele registro
 - acessa-se diretamente o registro seguinte

31

Registros de Tamanho Variável

- **Índice Externo:**
 - Armazena o byte offset de cada registro
 - deslocamento relativo ao início do arquivo
 - também usado para calcular o tamanho dos regs.
 - Através do índice:
 - **é possível acessar os registros diretamente**
 - mas isso demanda a leitura do arquivo de índice
 - sabe-se onde termina aquele registro

32

Registros de Tamanho Variável

Registro iniciados por indicador de tamanho:

```
27Maria|Rua 1|123|São Carlos|25João|Rua A|255|Rio Claro|27Pedro|Rua 10|56|Rib. Preto|
```

Arquivos de dados + arquivo de índices:

```
Dados: Maria|Rua 1|123|São Carlos|João|Rua A|255|Rio Claro|Pedro|Rua 10|56|Rib. Preto|
```

```
Índice: 00 27 52
```

33



Exercícios

- Seja o seguinte conjunto de campos:

Número: _____
Origem: _____ Destino: _____
Data: ___ / ___ / ___ Horário: _____ : _____
Poltrona: _____ Preço: _____

- Considere que campos como estes acima serão armazenados em um arquivo como uma seqüência organizada de caracteres.
- Dê 3 exemplos de realização desses campos, organizados via:
 - comprimento fixo, indicador de comprimento, delimitadores e tags

34



Bibliografia

- M. J. Folk and B. Zoellick, *File Structures: A Conceptual Toolkit*, Addison Wesley, 1987.**

39