

Coleções e Estruturas de Dados

Estruturas de Dados

- Formato de gerenciamento de informação
- Define organização interna dos dados, e operações sobre eles
 - Geralmente operações mantêm propriedades interessantes
- Projetadas para serem “eficientes”
 - Mas sacrifícios as vezes precisam ser feitos

Exemplo: String

- Uma string é apenas uma forma de representar texto
- Armazena texto internamente como um array contíguo de caracteres
- Terminado com o caractere especial '\0'
- Possui várias operações
 - strlen, strcpy, strcmp

Qual estrutura usar?

- A estrutura representa uma parte julgada relevante do todo
- Sempre algum lado sai perdendo
 - Você pode sacrificar eficiência de armazenamento por velocidade
 - Ou o contrário
 - Ou até modelar seu objeto de interesse de forma diferente
- Pode depender do algoritmo que for usado

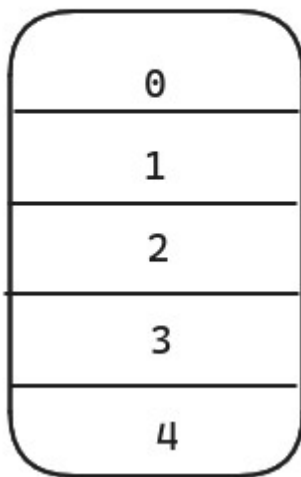
Exemplo hipotético: Rede Social

- Você pode querer modelar uma rede social de várias formas
 - Apenas nomes e fotos
 - Relações entre usuários
 - Listas de mensagens entre usuários
- Cada modelagem terá seus casos de uso, dependendo do seu objetivo

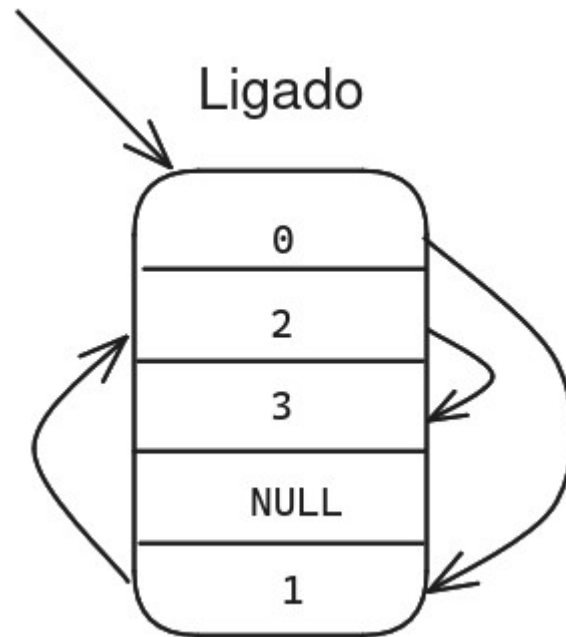
Organização dos dados

- Podem ser sequenciais, ou ligados.

Sequencial



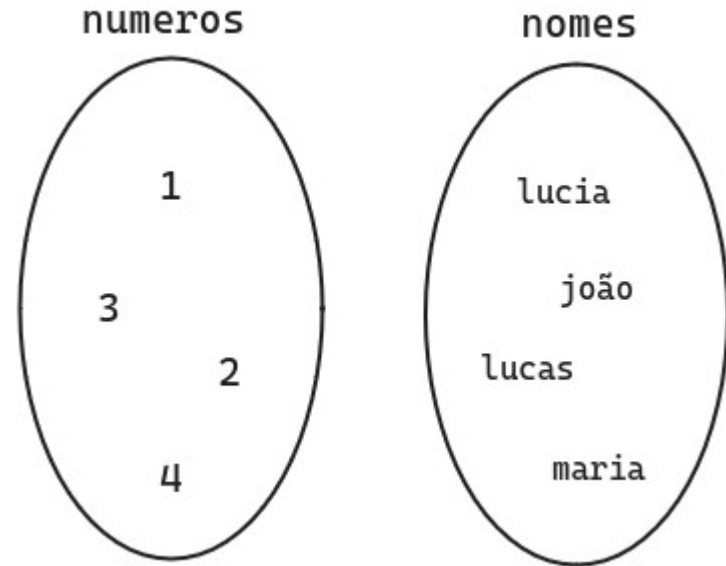
Ligado



Coleções

Conjuntos

- Definem um “domínio”
- Coleção de entidades similares
- Sem ordem
- Tipos (abstratos) representam conjuntos, instâncias representam elementos

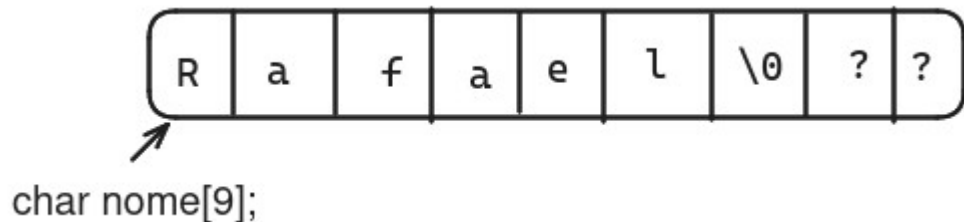


Sequencias

- Conjuntos ordenados do mesmo tipo
- Indexáveis e enumeráveis
- Equivalentes a arrays em C
- Teoricamente podem ser infinitos (porem sempre enumeráveis)

Exemplo: String

- Uma string em C é apenas um array (sequência) de caracteres
- Possui espaço máximo finito
- Por decisão de implementação, terminada sempre com '\0'



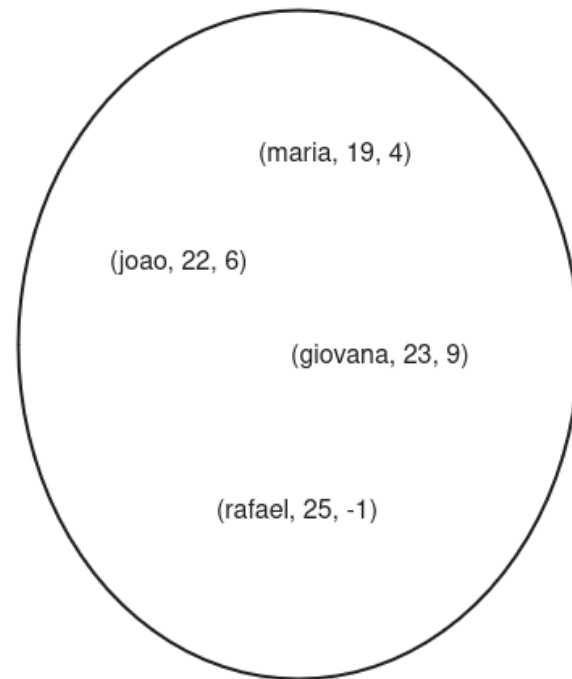
Tupla/N-Tupla

- Uma coleção de elementos de tipos possivelmente diferentes
- Tipos sempre seguem a mesma ordem, ex: (nome, idade, altura)
- Número fixo de elementos pra todas tuplas similares (ex: 3-tupla)
- Em C, ela pode ser representada por um struct
 - Structs ocupam um espaço contíguo de memória

Exemplo: Aluno

- O **conjunto de tuplas** na forma (nome, idade, período) pode ser representada assim:

```
struct aluno {  
    char nome[50];  
    int idade;  
    int periodo;  
};
```



- Note que nome é uma sequência.

Alguns pontos importantes

- Classes podem definir estruturas, mas estruturas não são apenas classes
 - Podem ser, por exemplo, coleções de structs e funções
- A organização interna dos dados só é relevante pra quem fez
- Estruturas de dados geralmente misturam os conceitos
 - Tanto juntando tuplas e sequencias, quanto utilizando referências
 - Veremos referências melhor em grafos

Acabou

<Imagem de Interrogação>