

Fundamentos de Algoritmos e Estrutura de Dados

Prof. André Gustavo Hochuli

gustavo.hochuli@pucpr.br
aghochuli@ppgia.pucpr.br

Plano de Aula

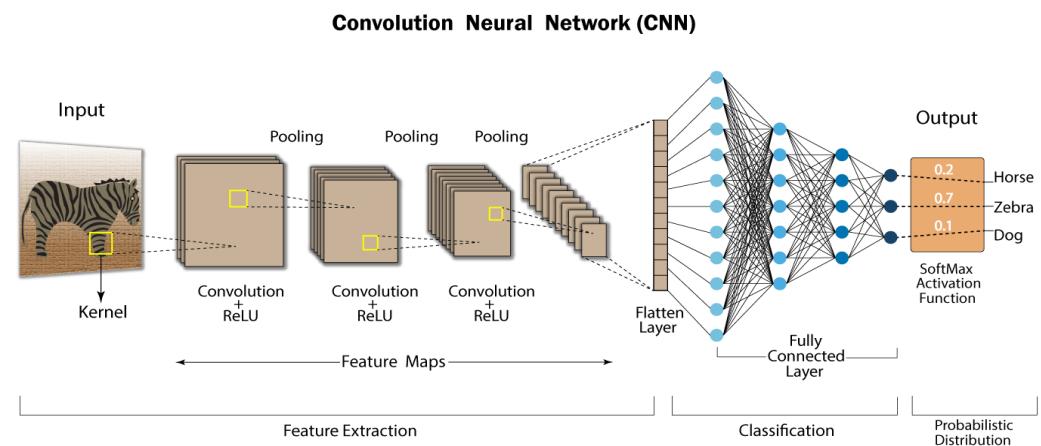
- **Apresentação do Professor**
- **O que esperar da disciplina?**
- **Plano de Ensino**
- **Ferramentas**

Prof. André Gustavo Hochuli

- Formação
 - Ciência da Computação [2004, PUCPR]
 - Mestre [2007, PPGIA/PUCPR]
 - Doutor [2018, PPGINF/UFPR]
- Experiência Profissional
 - P&D em Visão Computacional [2008-2013]
 - Professor Universitário [2014 - Atual]
- Linhas de Pesquisa
 - Aprendizagem de Máquina e Reconhecimento de Padrões



Hobbies:
Aviação
Futebol
Tecnologia

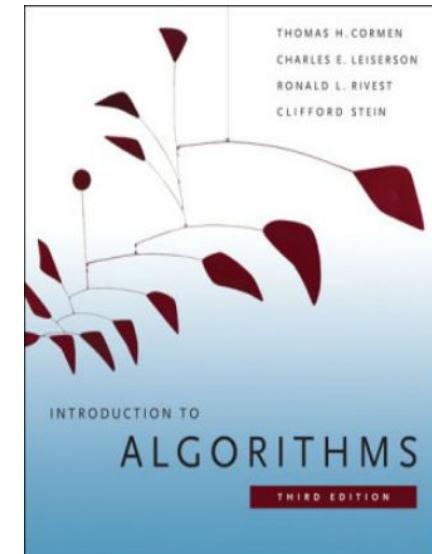


O que esperar da disciplina ?

- Abordagem Teórico – Prática
- Principais Tópicos em Estrutura de Dados
- Material em Inglês - Livros e Artigos
- Resolução de problemas do cotidiano
- Espaço para o estudante debater e trazer problemas/dúvidas
- Trabalhos em grupos
- Linguagens C e Python

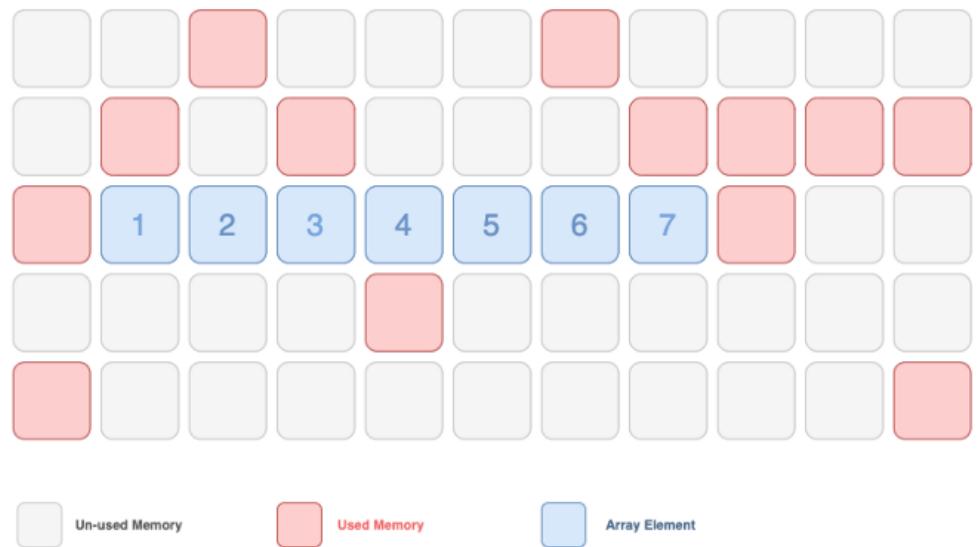
Plano de Ensino

- Introdução e Conceitos Básicos
- Métodos de Ordenação
- Estruturas de Dados Elementares (listas, pilhas, filas)
- Tabela Hash (Hashing)
- Árvores (Binária e AVL) e Métodos de Balanceamento
- Grafos
- Complexidade e Programação Dinâmica
- Avaliação:
 - Trabalhos, Seminários, Avaliações (Provas)
 - Trabalho de Recuperação
- Média: (Trab + Aval) / N
 - $N = \text{Qtd Trab} + \text{Qtd de Aval}$



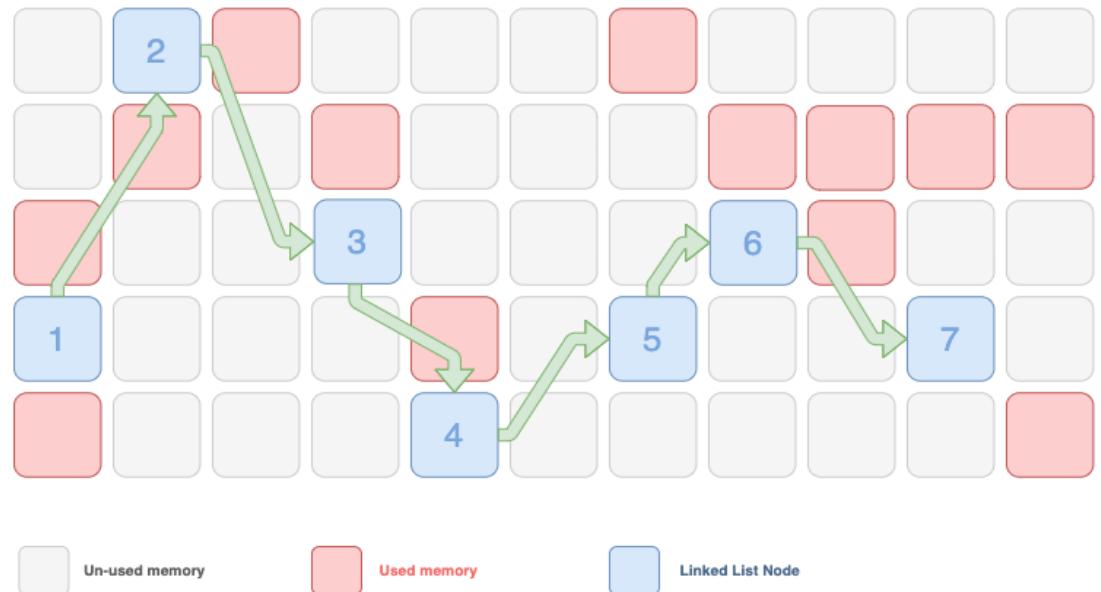
Introdução

- Arrays não apresentam bom desempenho ao armazenar grandes massas de dados e/ou dados complexos
- Alocação contígua
- Vantagens
 - Acesso é rápido e sequencial
 - Baixo Overhead
 - Requer baixo nível de programação
- Desvantagem
 - Inviável para grandes massas de dados
 - Limitado ao número de blocos sequenciais livres



Introdução

- Estrutura de dados utilizam sequências de ponteiros para blocos de memória
- Alocação não-contígua
- Vantagens
 - Armazenar grandes massas de dados
 - Memória física é o limite
- Desvantagem
 - “Desempenho”
 - Alto Overhead
 - Requer alto nível de programação



Estruturas Comuns

- Conceito de alocação não-contígua

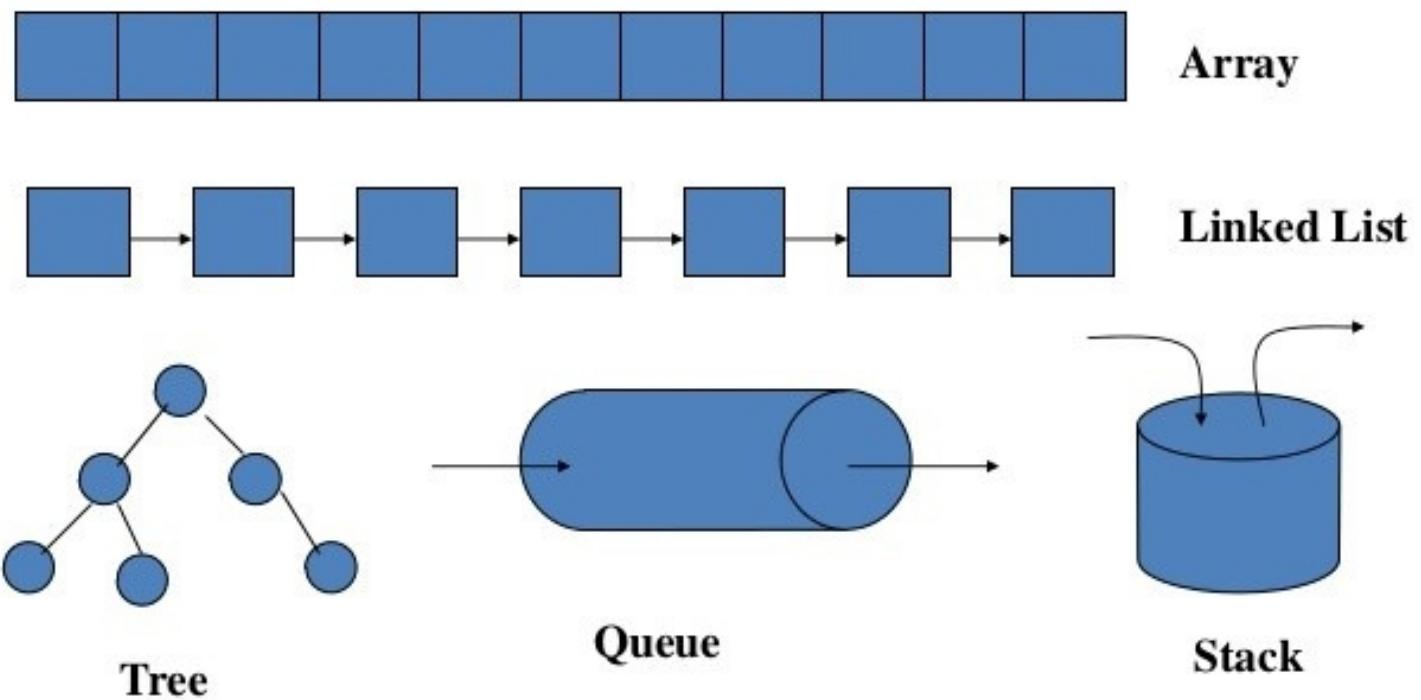
- Topologias

- Listas

- Pilhas
 - Filas

- Árvores

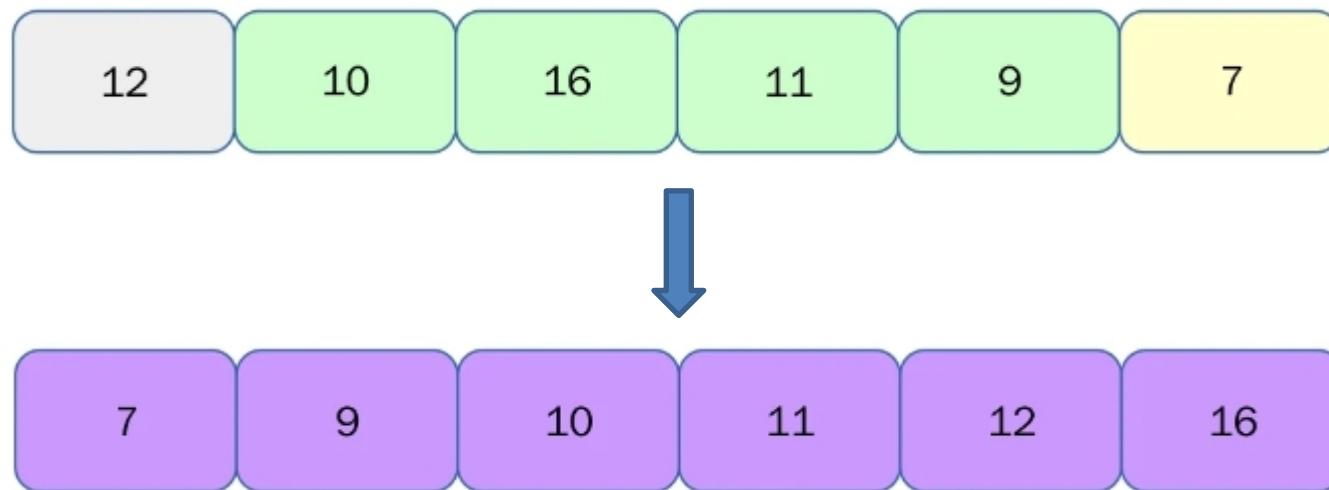
- Binárias
 - Red Black
 -



Ordenação e Busca de Dados

Ordenação

- Organizam os dados
- Melhora a performance de uma busca



Ordenação

- Bubble Sort

- Varre o vetor comparando pares, trocando se o elemento posterior for menor que o anterior (ordem descrecente)

```
def bubble_sort(arr):
    n = len(arr)
    for i in range(n):
        # Flag para otimização: Se nenhum elemento for trocado, o vetor já está ordenado
        troca = False
        for j in range(0, n - i - 1):
            if arr[j] > arr[j + 1]:
                # Troca os elementos
                arr[j], arr[j + 1] = arr[j + 1], arr[j]
                troca = True
        # Se não houve trocas, o vetor já está ordenado
        if not troca:
            break
    return arr

arr = [23, -2, 30, -20]
arr = bubble_sort(arr)
```

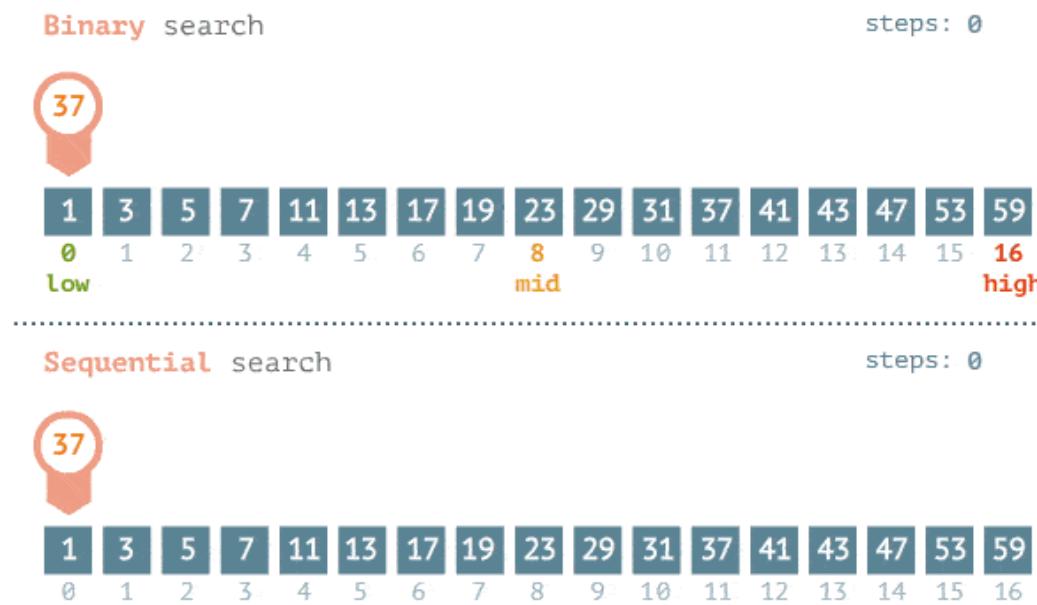
Ordenação

- Outros métodos:

- Bubble Sort
- Insert Sort
- Selection Sort
- Merge Sort
- Quick Sort
- Shell Sort
- Heap Sort

Métodos de Busca

- Encontrar um elemento em um conjunto de dados
- Sequencial ou Linear (arrays ordenados ou não)
- Binária (arrays ordenados)



www.penjee.com

Metódos de Ordenação e Busca (Trabalho)

- Trabalhos sobre algoritmos de ordenação e busca
- Comparação de Crítica de Métodos
- Formalização no Ambiente Virtual da Disciplina (AVA)

=> CRIAR GRUPOS DE TRABALHOS NO AVA <=

LET'S CODE!!



Para a próxima semana....

- **Pesquisa sobre as estruturas de dados Lista, Fila e Pilha.**
- **Leitura Recomendada:**
- **<http://117.211.166.170:8080/jspui/bitstream/123456789/1552/1/Linked.pdf>**
- **Bons estudos e boa semana!**