

SCC0602 - Algoritmos e Estruturas de Dados I

Apresentação



Professor: André C. P. L. F. de Carvalho, ICMC-USP
PAE:
Monitor:

© André de Carvalho - ICMC/USP

1

Aula de hoje

- Introdução
- Nossa postura
- Objetivos
- Curso
- Avaliação
- Bibliografia e Ferramentas

© André de Carvalho - ICMC/USP

2

Objetivo no Jupyter

- Apresentação de conceitos avançados que levem o aluno a uma maturidade em programação estruturada, com conhecimento de uma linguagem de programação com recursos avançados.
- Aprendizado de técnicas para construção de algoritmos e para análise da complexidade de algoritmos.
- Aprendizado de algoritmos clássicos de ordenação e busca em memória interna.
- Prática de Programação.

© André de Carvalho - ICMC/USP

3

Programa

- Conceitos avançados de análise de algoritmos.
- Paradigmas de projetos de algoritmos. Métodos de ordenação.
- Espalhamento (hashing).
- Grafos.

© André de Carvalho - ICMC/USP

4

Programa

- Conceitos avançados de análise de algoritmos:
 - Método da árvore de recorrência e teorema mestre.
- Paradigmas de projetos de algoritmos. Métodos de ordenação diretos e avançados:
 - Inserção, seleção, bubblesort, quicksort, heapsort.

© André de Carvalho - ICMC/USP

5

Programa

- Métodos de busca em memória interna: sequencial, binária e árvores, comparação entre métodos.
- Espalhamento (hashing).
- Grafos: noções básicas, representação, percurso e algoritmos clássicos sobre grafos (caminhos mínimos, ordenação topológica, componentes fortemente conectados, árvores geradoras mínimas).

© André de Carvalho - ICMC/USP

6

Etiqueta em sala de aula

- Chegar no horário da aula
- Pedir licença para entrar e sair da sala
- Usar palavras "mágicas" por favor, com licença, obrigado (a) e desculpe
- Não conversar durante a aula
- Levantar o braço para fazer perguntas e comentários
- Não ler outro material durante a aula
- Desligar celular durante a aula
- Colocar lixo no lixo
- Não copiar de colega ou internet material a ser avaliado

Código de Honra

- Espera-se que cada aluno obtenha seu grau baseado exclusivamente na avaliação de seu esforço e trabalho pessoal
 - Qualquer forma de conversa em exames, ou plágio em trabalhos é uma fraude
- Comportamentos que não são aceitos na USP
 - Atitudes previstas no Código Disciplinar Discente desta Universidade e são puníveis de acordo com o grau de severidade de acordo com esta regulamentação
- O código de honra será ser rigorosamente seguido

Andamento

- Aprender a aprender
 - Promover a cultura do conhecimento, ao invés de meramente transmitir informação
- Professor é um facilitador
 - Aluno precisa ler material do curso e praticar programação
 - Desempenho do aluno depende de sua dedicação

Material e Laboratório

- Material estará disponível no e-disciplinas
- Aluno deve ter estudado os conceitos vistos na aula teórica
 - Não se espera que o aluno pergunte sobre conceitos que estão nos slides
- Só se aprende praticando

Avaliação

- Provas:
 - Haverá duas provas
 - 11/4 e 20/6
 - Cada prova vale de 0 a 10
- Trabalhos:
 - Curso terá vários trabalhos práticos
 - Cada trabalho vale de 0 a 10

Avaliação

- Cálculo da média:
 - MP = Média Aritmética das Provas
 - MT = Média Aritmética dos Trabalhos
 - MF = Média Final
 - Se $MP \geq 5$ e $MT \geq 5 \rightarrow MF = (7MP + 3MT) / 10$
 - Se $MP < 5$ ou $MT < 5 \rightarrow MF =$ menor valor entre MP e MT
- Fraude: média final zero

Avaliação

- Recuperação:
 - Só terão direito à recuperação os alunos com $3.0 \leq MF \leq 5.0$ e frequência superior a 70%

Sobre o Curso

- Livro Texto:
 - CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. Algoritmos: Teoria e Prática. Editora Campus. 2002.
 - ZIVIANI, N., Projeto de algoritmos, 2a. Ed., Thomson, 2004.
- Bibliografia Complementar:
 - ROBERTS, E., Programming Abstractions in C, Addison Wesley, 1996.
 - SEDGEWICK, R., Algorithms in C, Addison-Wesley, 1990.
 - KERNIGHAM, B. W.; RITCHIE, D. M. C, A Linguagem de Programação Padrão ANSI, Editora Campus, 1995.

Conclusão

- Como vai ser o curso
- Objetivos e conteúdo
- Avaliação
- Material bibliográfico

Perguntas

