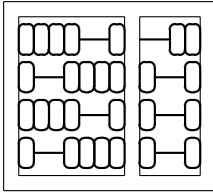


MC202



Estrutura de Dados

Prof. Neucimar J. Leite

Monitor: Felipe A. Louza

Auxiliar PAD: Ivan Henrique C. Petrin

IC – UNICAMP

Descrição da Disciplina

2º Semestre – 2014

Horário das aulas

Turma	Dia	Horário	Sala
CD	3ª	10 – 12	CB-02
CD	5ª	10 – 12	CB-03
CD	3ª	14 – 16	CC-02, CC-03

Ementa: Estruturas básicas para representação de informações: listas, árvores, grafos e suas generalizações. Algoritmos para construção, consulta e manipulação de tais estruturas. Desenvolvimento, implementação e testes de programas usando tais estruturas em aplicações específicas.

Atendimento: Dúvidas deverão ser tiradas logo após a aula. Não haverá atendimento no dia que precede à prova, assim como no dia da prova.

Exercícios: Durante o desenvolvimento do curso poderão ser indicados exercícios práticos e teóricos. Não será exigida a sua entrega, mas os conhecimentos adquiridos durante a resolução dos exercícios serão cobrados nas provas.

Monitoria: Esta disciplina conta com a participação do pós-graduando, Felipe Louza, que atuará principalmente nas atividades práticas de laboratório, e do monitor (PAD) Ivan Petrin.

Avaliação: A avaliação da disciplina será baseada nas notas de três provas teóricas P1, P2 e P3 e na média ponderada L das notas de projetos de laboratórios. Cada projeto de laboratório terá um peso específico, variando de acordo com a complexidade da tarefa solicitada. Alguns laboratórios deverão ser feitos no prazo de uma semana e outros no de duas semanas. Os pesos e os prazos dos laboratórios serão divulgados juntamente com os enunciados dos laboratórios. Só serão considerados para avaliação os programas considerados corretos pelo sistema automático utilizado, ou seja, aqueles que passarem por todos os casos de testes. Os programas aprovados por este sistema serão avaliados manualmente levando-se em conta aspectos como correção, comentários, clareza, qualidade e eficiência.

A média das provas P é calculada da seguinte forma:

$$P = (3 * P1 + 3 * P2 + 4 * P3) / 10$$

A nota final do semestre, antes do exame, M será calculada como a média harmônica entre as médias das provas (P) e a média dos laboratórios (L), da seguinte forma:

- $M = (2 \cdot P \cdot L) / (P + L)$, se $(P + L) > 0$,
- $M = 0$, se $P = L = 0$.

O aluno terá direito a fazer o exame se $2.5 \leq M < 5$.

A nota final da disciplina (F), após o exame (E), será calculada pela fórmula:

- $F = \min\{5, (M + E)/2\}$, se $2.5 \leq M < 5$ e o aluno compareceu ao exame,
- $F = M$, caso contrário.

Se $F \geq 5$, o aluno será considerado aprovado na disciplina. Caso contrário, será considerado reprovado.

As provas e o exame final serão realizados nos dias indicados a seguir, nos horários correspondentes às aulas normais:

- 2/10: primeira prova P1
- 11/11: segunda prova P2
- 18/12: terceira prova P3
- 15/01/2015: Exame

Observações:

1. Qualquer tentativa de fraude nas provas e projetos de laboratório implicará em aproveitamento zero na disciplina para todos os envolvidos.
2. Não haverá provas substitutivas.
3. Quaisquer alterações referentes a procedimentos e datas acima poderão ser implementadas levando-se em conta uma votação por maioria simples dos alunos em sala de aula. Estas eventuais alterações serão divulgadas no *web site* do curso.

Programa:

Os itens apresentados a seguir indicam os tópicos a serem discutidos durante o semestre:

1. Representação de matrizes por linearização de índices; acesso por linhas e colunas.
2. Estruturas ligadas: nó, apontador, variável apontadora, alocação dinâmica de memória.
3. Listas ligadas simples: operações básicas.
4. Comparação de listas ligadas com vetores.
5. Algoritmos gerais para listas simples: enumeração, inversão, cópia, concatenação.
6. Pilhas, filas, e aplicações (inclusive eliminação de recursão).
7. Intercalação (merge) de listas e mergesort; análise informal.
8. Variações: listas circulares, duplamente ligadas, com cabeça. Lista livre.
9. Árvores binárias: representação e percurso (recursivo).
10. Aplicação: árvores de busca (com inserção e remoção).
11. Fila de prioridade (heap) implementação com vetor e heapsort.
12. Árvores gerais: definição, representação por listas, percursos.
13. Listas generalizadas e uso para representar estruturas ligadas em geral.
14. Árvores B e generalizações.
15. Introdução ao espalhamento (hashing): conceito, implementação com listas ligadas. Técnicas de espalhamento para arquivos
16. Grafos: conceito, representação por matrizes e listas ligadas. Percurso de grafos em largura e profundidade.

A linguagem de programação a ser utilizada nos exemplos discutidos em aula será C.

Bibliografia indicativa

1. N. Ziviani, *Projeto de Algoritmos com Implementações em Pascal e C* (2a. ed.). Thomson (2004). ISBN 85-221-0390-9.
2. J. L. Szwarcfiter e L. Markenzon. *Estruturas de Dados e Seus Algoritmos*. Editora LTC (1994).
3. R. Sedgewick, *Algorithms in C*. Addison-Wesley (1990). ISBN 0201514257.
4. A. M. Tenenbaum, Y. Langsam, e M. J. Augenstein. *Estruturas de Dados Usando C*. Makron Books (1995).
5. A. V. Aho, J. E. Hopcroft, e J. Ullmann. *Data Structures and Algorithms*. Addison-Wesley (1983).
6. D. E. Knuth, *The Art of Computer Programming, Vol I: Fundamental Algorithms*. Addison-Wesley (1978).
7. E. M. Reingold e W. J. Hanson, *Data Structures*. Little-Brown (1983).
8. N. Wirth, *Algorithms + Data Structures = Programs*. Prentice-Hall (1976).