

ALGORITMOS E LINGUAGEM DE PROGRAMAÇÃO

AVALIAÇÃO

GRUPOS de LAB: Até 4 ALUNOS

MÉDIA: MÉDIA_LAB – 2pts - entregar no dia da prova semestral oficial

Teste - 2pts - simulado ENADE

PROVA_SEMESTRAL (oficial) - 6pts – Final do semestre

LIVRO TEXTO:

Treinamento em Linguagem C, Victorine Viviane Mizrahi, Pearson, Prentice-Hall, 2ª edição, São Paulo

BIBLIOGRAFIA ADICIONAL - - Qualquer outro livro / apostila de **LINGUAGEM C**.

Ementa / sequência - Ver livro texto – Capítulos 1 a 5, 7

ORIENTAÇÕES PARA CONFEÇÃO DOS RELATÓRIOS DE LABORATÓRIO

As práticas de laboratório visam, não a simples programação mas o entendimento completo do problema e sua solução.

O aluno deve ser capaz, não apenas de realizar a programação, mas interpretar, de forma crítica os resultados obtidos. Ele deve ser capaz de obter uma solução analítica ou numérica aproximada, se possível, para que possa comparar os resultados e verificar a correção das saídas de seu programa.

Para cada problema resolvido, o relatório deve conter os seguintes itens:

1) DEFINIÇÃO DO PROBLEMA - FOLHA COM O ENUNCIADO original do problema

2.) SOLUÇÃO

- Detalhes adicionais necessários para definir completamente o problema a ser resolvido
- Explicar qual, porque, como, e a razão da escolha da solução.
- Apresentar uma solução teórica/gráfica, etc, não computacional, para fins de comparação com a solução computacional.
- Apresentar algoritmo da solução computacional.

3) VALIDAÇÃO

- Detalhar como pretende realizar os testes para verificar o correto funcionamento do programa
- Demonstrar a obtenção dos resultados com as sequências de teste – via simulação manual e dados de entradas versus saídas.

3.) COMENTÁRIOS E CONCLUSÕES

- Análise dos resultados:
 - * Conformidade com o resultado teórico / esperado.
 - * Discrepâncias.
 - * Explicações.
- Aspectos relevantes de implementação.
- Problemas encontrados.
- Casos e testes realizados para verificação de funcionamento.
- sugestões de melhoria e /ou correções.

4.) LISTAGENS

- Código comentado, estruturado, em linguagem C.
- Resultados, saída (via print-screen ou saída direta na impressora).
- Sequências de testes e suas saídas.

OBS: Software = programa + documentação

Diretrizes para a Documentação de Programas [UNICAMP]

*Texto retirado da Apostila do Curso MC102 – Algoritmos e programação de computadores, UNICAMP, 2009, dos professores **Luiz Gustavo Turatti** (PED-A), **Felipe B. Valio** (PED-C), **Júlio César F. Cornacchia** (PAD), Versão 1.0.0 b – 05.out.2009*

A documentação de um programa complementa o seu código para torná-lo mais compreensível para quem for analisar os detalhes de sua implementação bem como para facilitar o entendimento de seu funcionamento. Pretende-se, com a cobrança de documentação, criar em você o hábito de registrar, ainda que de forma mínima, detalhes complementares importantes relativos à construção dos programas por você produzidos bem como decisões de projeto tomadas por você durante a sua implementação. Na sua vida profissional, dificilmente você irá desenvolver software apenas por você. Além de embutir no código informações complementares para o seu entendimento e de outras pessoas que necessitem entender a forma como o programa em questão foi concebido e construído, é preciso deixar claro, na interface de usuário, em que estado o programa em execução se encontra e o que é esperado do usuário em cada estado.

A documentação também é importante para você próprio. Depois de um certo tempo não lembramos mais de decisões de projeto tomadas quando da confecção de um determinado software. Uma documentação de um software deve registrar estas decisões para facilitar a manutenção deste software. A documentação, portanto, é tão importante quanto o código. Por esta razão, documente bem aquilo você produz!

A documentação de um programa pode estar embutida no próprio programa e/ou ser fornecido na forma de documentos complementares, como um manual de usuário. Na disciplina, a exigência mínima é um código bem comentado e bem estruturado. É muito importante que você se preocupe com a apresentação do código do programa de modo a facilitar sua leitura. As linhas no código devem ser alinhadas de modo a refletir a estrutura do programa. Comandos internos a outro devem ser deslocados à direita.

Comentários devem ser colocados em pontos estratégicos, presentes em todas as partes onde o código do programa não é trivial ou onde o papel de alguma variável não seja de compreensão imediata. Não comente o óbvio! Além desse tipo de comentários elucidativos, no mínimo os seguintes comentários devem ser inseridos em cada programa produzido por você:

- Informações de caráter administrativo:
 1. Nome e RA do autor do programa.
 2. A identificação da atividade a que se refere o programa.
 3. Data em que foi produzido o programa.
- Descrição informal e sucinta do que o programa faz. Esta descrição deverá ser feita, de maneira sintética, e não deverá transcrever literalmente, comando por comando, em Português.
- Explicações sobre como operar o programa, ressaltando quais são as entradas, como serão solicitadas e quais são as saídas esperadas.
- Breve descrição dos algoritmos utilizados na resolução do problema proposto. Aqui cabe novamente a observação acima. Não transcreva para o Português o seu código em C. Abstraia detalhes de implementação. Redija descrições em nível mais conceitual. Algoritmo e programa são coisas distintas. Um algoritmo serve de base para uma implementação de um software sendo que a partir de um algoritmo podemos derivar diversas implementações alternativas.
- As condições de contorno, ou seja, qual é o conjunto válido de dados de entrada para o correto funcionamento do programa. Em que situação o seu programa não funcionará