

Trabalho 1 – especificação

1. Objetivo

Empregar os conhecimentos adquiridos sobre estruturas de dados (TAD, listas, pilhas e filas) no desenvolvimento de uma aplicação computacional, seguindo um conjunto de especificações funcionais e de projeto, assim como boas práticas de programação.

2. Indicações Gerais

O trabalho deve ser desenvolvido em grupos de dois alunos, no máximo, sendo que:

- Cada grupo deverá fazer o trabalho completo em suas duas versões (estática e dinâmica).
- Serão aceitos somente trabalhos em linguagem de programação C sequencial (sem uso de GOTO).
- **Data de entrega: 01/outubro (Domingo), até meia noite.**

3. Critérios de Avaliação

Os trabalhos serão avaliados de acordo com os seguintes critérios:

- Usabilidade da interface: a interface com o usuário deve ser clara e intuitiva, mesmo que seja via linha de comando, exibindo mensagens claras para o usuário.
- Corretude do programa: o programa deve fazer o que foi especificado.
- Estruturas de dados utilizadas: adequação e eficiência.
- Observação dos “bons modos” da programação: TAD, modularidade do código, documentação interna, indentação, etc.
- A cada dia de atraso, 1 ponto é descontado da nota. Lembrem-se de que a média final dos trabalhos deve ser maior ou igual a 5 para que o aluno seja aprovado
- O plágio de programas não será tolerado. Quaisquer programas similares terão nota zero, independentemente de qual for o original e qual for a cópia.

4. Entrega do Trabalho

A entrega dos trabalhos será na atividade aberta no Tidia. Serão requeridos:

- Um arquivo comprimido (RAR, ZIP, etc.), que deverá ser postado por 1 dos membros do grupo no site da disciplina, e deve conter (a) arquivos de código-fonte do programa (todos .c e .h, necessários para o programa), e (b) relatório breve. O relatório, de aproximadamente 5 páginas, deve conter, pelo menos, (i) identificação dos integrantes do grupo, (ii) breve descrição do trabalho feito, (iii) apresentação das estruturas de dados utilizadas e justificativas necessárias e (iv) avaliação sobre os procedimentos mais caros computacionalmente para o uso do sistema.
- Todos os arquivos enviados (relatório, .c e .h) devem conter o nome e NUSP de ambos integrantes da dupla

5. Especificações

Nesta seção, são apresentadas as especificações funcionais do programa que será desenvolvido.

- Seguindo as características pedidas abaixo, faça duas implementações da estrutura de dados (usando TADs), sendo que uma deve ser **totalmente estática** e a outra deve ser **totalmente dinâmica**.

- Para poupar memória, a versão estática deve ser feita, **obrigatoriamente**, utilizando banco de memória.
- O programa principal deverá usar apenas um dos dois TADs de cada vez: teste uma vez com um TAD, depois compile novamente para testar o segundo TAD, SEM ALTERAR QUALQUER COMANDO NO PROGRAMA PRINCIPAL (exceto o nome do arquivo do TAD no include: o arquivo .h). Na entrega, o include deve ser da versão estática, com o include da versão dinâmica comentado.
- Para questões específicas que não estejam especificadas, use o bom senso e documente suas decisões no relatório a ser entregue. Caso necessite, contate o estagiário PAE ou o professor da disciplina para tirar eventuais dúvidas.

Descrição do trabalho: Sistema de controle de biblioteca

Desenvolva um sistema de controle de uma biblioteca, incluindo toda a interface para interação com o usuário, permitindo cadastrar, retirar e devolver livros. O sistema deve ser desenhado para ser utilizado diretamente por uma bibliotecária, sem acesso remoto ou direto pelos alunos.

O sistema precisa realizar pelo menos as seguintes operações:

- a) Cadastrar aluno, contendo nome, NUSP, telefone e e-mail. Cuidado com os limites de inteiro para o número de telefone.
- b) Cadastrar livro, contendo título, autor, ISBN, editora, ano, edição, etc. Considere que cada livro pode ou não estar disponível e que várias cópias do mesmo livro podem existir.
- c) Retirar livro, se disponível o livro deve ser retirado pelo aluno determinado, se não o aluno deve entrar numa fila de espera.
- d) Retornar livro, o próximo aluno na lista de espera é indicado ou, se a fila estiver vazia, o livro fica disponível.
- e) Remover livro, remove o livro da biblioteca.
- f) Remove aluno, remove o aluno cadastrado e retira ele de TODAS as filas de espera.

Outras operações podem ser necessárias para construção do sistema, considerem usabilidade e funcionalidade no desenvolvimento. Por exemplo, considere que para tentar retirar o livro o sistema precisa conter algum sistema usável de busca.

Por último, o sistema deve conter um método para envio de mensagens para os usuários (para quando o a vez dele chegar ou caso alguma informação precise ser trocada). As mensagens devem ser guardadas em um PILHA de mensagens, assim como um sistema de e-mail. Como o sistema só tem um usuário, deve existir algum método para imprimir todas as mensagens de um aluno, na ordem da pilha, acessível pelo usuário.