

F2 – TAD Pilha: implementação, teste e aplicação

Atividade avaliativa para cômputo de Frequência

Descrição

- Implemente uma **Pilha com alocação sequencial e alocação estática** de memória;
- Implemente a Pilha como um **Tipo Abstrato de Dado**, em um **arquivo separado** do programa principal; inclua **no TAD Pilha apenas as Operações Primitivas** de uma Pilha; o arquivo contendo o TAD Pilha deve ser uma **unidade de software independente**, ou seja, deve conter exclusivamente o TAD Pilha;
- Implemente um **programa principal para testar o TAD Pilha**;
- No programa principal, ou em outra unidade de software, caso preferir, **implemente pelo menos duas operações não primitivas** sobre Pilhas, que podem ser, por exemplo, **imprimir / mostrar** todos os elementos da Pilha, retornar o valor do elemento do topo, retornar o número de elementos da pilha, verificar se 2 pilhas são iguais, ou algum outro exercício indicado no livro Estruturas de Dados com Jogos ou em algum outro livro. Desenvolva essas operações “sem abrir a TV”, ou seja, **manipulando o TAD Pilha exclusivamente através dos operadores primitivos**. Busque proporcionar a maior independência e portabilidade de código que for possível;
- **Descreva uma aplicação de Pilhas**. Indique como a estrutura de dados seria utilizada para viabilizar essa aplicação;
- Dado que desenvolva o solicitado acima, fique à vontade para implementar outras funcionalidades e/ou personalizar sua solução, caso desejar; é possível implementar uma “Pilha Burra” ou uma “Pilha Inteligente” (Exercícios 2.12 e 2.15 do livro Estruturas de Dados com Jogos”);
- Desenvolva o trabalho individualmente;
- Utilize **C ou C++**.

O que entregar (em **arquivo único – PDF, sem compactar**)

- 1- Código fonte;
- 2- Documentação contendo nome, RA, desenho das estruturas utilizadas, prints da execução e o que mais você julgar necessário.

Quando entregar

- Até o dia anterior à próxima atividade síncrona.

Onde entregar

- No ambiente de interação da disciplina no Google Classroom, no link indicado.

Possíveis problemas (usar como um checklist)

- 1- Trabalho incompleto ou diferente do especificado;
- 2- Erro nos algoritmos;
- 3- Não entregar código-fonte ou não entregar documentação contendo nome, RA, diagrama das estruturas utilizadas e prints da execução;
- 4- Implementação incompatível com o conceito de tipos abstratos de dados (não usar arquivos separados para módulos independentes; manipular dados do TAD sem ser pelos operadores primitivos; incluir no TAD operações não primitivas);
- 5- Não entregar no prazo indicado;
- 6- Trabalho desenvolvido/entregue em grupo (deve ser desenvolvido e entregue individualmente);
- 7- Plágio.