

Mini Projeto - Estruturas de Dados em C

Projeto Temático Livre com Implementação Obrigatória de Estruturas Lineares

Objetivo

Desenvolver, em linguagem C, um sistema computacional com tema livre definido pelos alunos. O foco principal é demonstrar domínio na modelagem de dados, uso de ponteiros, alocação dinâmica de memória e implementação manual de estruturas de dados clássicas.

Proposta

Cada equipe deverá escolher um problema real ou fictício, descrever o contexto do sistema e implementar um fluxo funcional com cadastro, inserção, remoção, consulta e processamento de informações. O tema é livre, porém todas as estruturas abaixo devem ser utilizadas obrigatoriamente.

Estruturas Obrigatórias

O projeto deve utilizar obrigatoriamente:

- struct (estrutura para modelagem das entidades do sistema)
- Array (vetor ou matriz)
- Ponteiros com alocação dinâmica (malloc/free)
- Lista encadeada
- Pilha (Stack – LIFO)
- Fila (Queue – FIFO)

Descrição de Uso das Estruturas

Struct: modelar a entidade principal do domínio (ex.: usuário, produto, paciente, tarefa).

Array: armazenamento estático, relatórios, histórico ou estatísticas.

Ponteiros: manipulação de memória dinâmica e nós das estruturas.

Lista Encadeada: coleção dinâmica com inserção e remoção durante a execução.

Pilha: representar comportamento LIFO (ex.: histórico, undo/redo, urgências).

Fila: representar comportamento FIFO (ex.: atendimento, processamento, requisições).

Requisitos Funcionais Mínimos

- Cadastro de registros
- Inserção nas estruturas
- Remoção de elementos
- Busca/consulta
- Processamento de dados
- Menu interativo no terminal

Restrições Técnicas

- Proibido utilizar bibliotecas prontas de estruturas de dados
- Todas as estruturas devem ser implementadas manualmente
- Uso obrigatório de malloc/free
- Código organizado, modularizado e comentado

Entregáveis

- Código-fonte (.c e/ou .h)
- Makefile ou instruções de compilação
- README contendo: tema escolhido, descrição do sistema, justificativa do uso das estruturas e instruções de execução
- Repositório do Github

Critérios de Avaliação

Uso correto das estruturas – 40%

Uso de ponteiros e memória dinâmica – 20%

Organização e modularização – 15%

Funcionalidade do sistema – 15%

Clareza e documentação – 10%

Exemplos de Temas (opcional)

Biblioteca, Estoque, Delivery, Hospital, Agenda, Sistema Acadêmico, Jogo, Logística, Rede Social, Smart City.

Resultado esperado: programa executável em terminal demonstrando domínio prático de estruturas de dados em C.