

Fundamentos de Programação

Ano/Semestre: 2024/1

Prova de Grau B

Instruções

Resolver as questões com consulta somente ao material apresentado em aula e aos códigos-fonte dos exercícios resolvidos pelo próprio aluno. **Os algoritmos não devem ser *hard-coded***, ou seja, com apenas uma solução fixa para o conteúdo disponível. Você deve enviar apenas os arquivos .c que você gerar. **NÃO ENVIE PROJETOS DO CODEBLOCKS OU EXECUTÁVEIS.**

Questões

1. (4,0 pontos). Desenvolva um sistema para gerenciar dados de sensores ambientais usando alocação dinâmica de memória. Primeiro, defina a struct “SensorData” com ponteiros para leituras de float, tamanho atual e capacidade. Em seguida, implemente a função “createSensorData” para inicializar a struct com uma capacidade inicial. Implemente a função “addReading” para adicionar leituras ao array, expandindo a capacidade em 50% quando necessário. Use realloc. Por fim, escreva “freeSensorData” para liberar a memória alocada.

Exemplos de leitura:

- `addReading(sensorData, 23.5);` // Adiciona uma leitura de 23.5°C.
- `addReading(sensorData, 22.0);` // Adiciona uma leitura de 22.0°C.

Definição da Struct:

- `float* readings;` // Array dinâmico para armazenar leituras de temperatura.
 - `int size;` // Número de leituras atualmente no array.
 - `int capacity;` // Capacidade total do array, indicando quantas leituras podem ser armazenadas antes da necessidade de expansão.
2. (6,0 pontos). Dado o arquivo anexo “veículos.txt”, contendo as colunas fabricante, modelo, ano, tanque e consumo (a primeira linha contém apenas o cabeçalho, ou seja, pode ser descartada), faça um programa que leia o arquivo e mostre na tela:
 - a) (1,5) Os fabricantes e modelos dos veículos fabricados entre 2015 e 2018, inclusive;
 - b) (1,5) Somente os modelos dos veículos que iniciam com vogal ou terminem com consoante;
 - c) (3,0) O modelo, ano e a autonomia dos veículos com menor e maior autonomia, calculada pela fórmula: $\text{autonomia} = \text{tanque} * \text{consumo}$.