

Notas da turma

Um professor da Fundação de Gessos e Alvenarias (FGA) está oferecendo a disciplina de Estruturação e Desenvolvimento de Alvenarias 1 (EDA-1). Como ele tem muitos alunos, o professor resolveu desenvolver um programa para a geração de diversos relatórios sobre os alunos. Para cada relatório, o professor constroi uma *lista encadeada* com os dados que interessam, e o objetivo é montar outras listas dividindo os dados de acordo com o que interessa ao professor ver no relatório em questão.

Como o professor entende bem de construção civil apenas, ele nos pediu ajuda para desenvolver esse algoritmo.

Tarefa

Considere uma lista encadeada composta de nós da forma

```
typedef struct celula {
    char matricula[10];
    double mediaAtividades;
    double mediaProvas;
    struct celula *prox;
} celula;
```

onde `matricula` contém a matricula do aluno, `mediaAtividades` contém a média das atividades do aluno ao longo do semestre e `mediaProvas` contém a média de provas.

Sua tarefa é escrever uma função

```
void relatorioMediaAtividades (celula *le, celula *l1, celula *l2, double media);
```

que recebe uma lista encadeada com nó cabeça `le` e devolve duas listas encadeadas com nós cabeça `l1` e `l2` de forma que todos os alunos na lista `l1` possuem média de atividades menor ou igual à `media` e na lista `l2`, média de atividades maior que `media`.

Você deve submeter um arquivo contendo

- Os `#include` necessários para que sua função funcione.
- A definição da `struct celula`.
- A função `relatorioMediaAtividades`.

Sua solução **deve cumprir os seguintes pré-requisitos**:

1. Você não deve alocar nenhuma nova célula na sua função, apenas manipular os ponteiros dos nós de `le` para que estejam em `l1` ou `l2`.
2. Você deve considerar que os nós cabeça `l1` e `l2` já foram alocados antes da chamada para a função `relatorioMediaAtividades`.
3. Como consequência, a lista encabeçada por `le` não estará intacta após a chamada à sua função.

Exemplo

Suponha, por exemplo, que a lista `le` seja

`le -> {140164006, 10.0, 8.0} -> {160016169, 4.0, 2.0} -> {170062465, 9.5, 8.5} -> {190262661, 5.0, 7.0} ->`

A chamada

```
relatorioMediaAtividades (le, l1, l2, 6.0)
```

deve devolver as listas

`l1 -> {160016169, 4.0, 2.0} -> {190262661, 5.0, 7.0} -> NULL`

e

12 -> {140164006, 10.0, 8.0} -> {170062465, 9.5, 8.5} -> NULL

Author: John L. Gardenghi