

EDFood: Pedidos

Professora Patrícia Dockhorn Costa, patricia.d.costa@ufes.br



Descrição

Você foi contratado para desenvolver um sistema que mantém um conjunto simples alimentos (veganos e não veganos) de um certo pedido. As seguintes funcionalidades são esperadas:

- Possibilidade de inserir alimentos veganos e não veganos;
- Organizar alimentos do pedido por número de calorias (CALORICOS ou MAGROS);
- Calcular o valor do pedido.

Arquivos Fornecidos e Esperados

Para facilitar, alguns arquivos da implementação já estão sendo disponibilizados, a saber:

Arquivos	Descrição
ComidaVegana.h e ComidaVegana.c	Definição e Implementação do Tad Comida Vegana
ComidaNaoVegana.h e ComidaNaoVegana.c	Definição e Implementação do Tad Comida Não Vegana
Pedido.h	Definição do Tad Pedido
Testador1.c	Testa funcionalidades básicas (vale até 6 pontos)
Testador2.c	Testa funcionalidades intermediárias (vale até 8 pontos)
Testador3.c	Testa todas as funcionalidades (vale até 10 ponto)

Você precisa implementar o Tad Pedido e a lista encadeada necessária para resolver o problema definido (faz parte da solução escolher o tipo de lista adequado). Ou seja, você precisa fornecer 3 arquivos, a saber: **Pedido.c**, **Lista.h** e **Lista.c**.

Importante:

- **Coloque seu nome em TODOS os arquivos que você produzir.**
- **A lista precisa ser encadeada!**
- **Use tipos opacos, como normalmente fazemos.**

Dicas

Funções de manipulação de strings <string.h>:

- `strdup`: aloca espaço para string e a copia (`char* destino = strdup(origem)`).
- `strlen`: Retorna o tamanho da string (sem o `'\0'`).
- `strcpy`: Faz a cópia de strings, desde que a string destino tenha tamanho suficiente. `strcpy (destino, origem)`.
- `strcmp`: Compara duas strings. `strcmp (str1, str2)`. Retorna:
 - 0 se str1 for igual a str2.
 - <0 se str1 for menor que str2 (como a sequência das palavras no dicionário)
 - >0 se str1 for maior que str2

Para compilar e gerar o executável

- `gcc -c <seunome>.c` (compila o arquivo e gera arquivo `<seunome>.o`)
- `gcc -c proglista1.c` (compila o arquivo e gera arquivo `proglista1.o`)
- `gcc <seunome>.o proglista1.o` (faz a linkagem e gera o executável `a.out`)