

ID: EO1-2025

## Problema

Um restaurante pediu a sua ajuda para desenvolver um gerenciador, capaz de fazer operações e relatórios envolvendo os alimentos preparados pelo restaurante. Desenvolva dois TADs envolvendo um tipo `tRestaurante` e um tipo `tAlimento`, bem como uma `main` para executar suas funcionalidades.

O tipo `tAlimento` possui um nome (`char[ ]`), um preço (`float`), uma quantidade em gramas (`float`) e uma quantidade consumida, também em gramas (`float`). O TAD para `tAlimento` acompanha as seguintes funções:

- **`criaAlimento(char *nome, float preco, float qtd)`**: inicializa um tipo `tAlimento` a partir dos parametros. Para mais especificações, verifique o `.h`.
- **`leAlimento()`**: le as informações de um `tAlimento` a partir da entrada seguindo a ordem: nome, preço, quantidade disponível
- **`consumirAlimento(tAlimento alimento, float qtd)`**: reduz a quantidade disponível de um `tAlimento`.
- **`rendaAlimento(tAlimento alimento)`**: calcula a renda produzida por um alimento, multiplicando a quantidade consumida pelo preço.
- **`imprimeAlimento(tAlimento alimento)`**: imprime na saída as informações de um tipo alimento, a partir da seguinte formatação:

---

Nome: pudim de arroz  
Quantidade: 1200 gramas  
Preço: R\$ 12.00/kg  
Quantidade consumida: 200 gramas  
Renda: R\$ 2.4

---

OBS: as linhas entre o texto são hífen, veja a saída nos arquivos `.h` para compreender melhor

O tipo `tRestaurante` possui um nome (`char[ ]`), número atual de alimentos (`int`) e uma lista de alimentos (`tAlimento[ ]`). O TAD para `tRestaurante` acompanha as seguintes funções:

- **`criaRestaurante(char *nome)`**: inicializa um tipo `tRestaurante` a partir dos parametros. Para mais especificações, verifique o `.h`.
- **`leRestaurante()`**: le as informações de um `tRestaurante` a partir da entrada, lendo o nome.
- **`cadastraAlimento(tRestaurante r, tAlimento a)`**: recebe um tipo `tAlimento` e o armazena na array de `tAlimento` de um tipo `tRestaurante`.
- **`consumirAlimentoRestaurante(tRestaurante r, char *nome, float qtd)`**: procura um tipo `tAlimento` dentro da array de `tAlimento` de um `tRestaurante` a partir do nome, e consome a quantidade `qtd` do alimento encontrado.

- 
- **quantidadeConsumida(tRestaurante r)**: retorna a quantidade total consumida por um restaurante, somando a quantidade consumida de todos os alimentos armazenados na array de tAlimentos de um tipo tRestaurante.
  - **rendaTotal(tRestaurante r)**: retorna a renda total de um restaurante, somando a renda produzida por todos os alimentos armazenados na array de tAlimentos de um tipo tRestaurante.
  - **void imprimeRestaurante(tRestaurante r)**: imprime as informações de um restaurante a partir da seguinte formatação:

Nome: Bicho Guloso

Alimentos:

---

Nome: pudim de arroz

Quantidade: 1200.00 gramas

Preço: R\$ 12.00/kg

Quantidade consumida: 200.00 gramas

Renda: R\$ 2.40

---

Nome: pudim de linguica

Quantidade: 2000.00 gramas

Preço: R\$ 21.00/kg

Quantidade consumida: 1000.00 gramas

Renda: R\$ 21.00

---

Quantidade total consumida: 1200.00 gramas

Renda total: \$ 23.40

Todas as funções e structs então incluídas nos .h disponibilizados, analise-os e crie os .c desenvolvendo as funcionalidades de cada função.

Além dos TADs, uma main deve ser implementada para testar as funcionalidades implementadas.

Na main, você começará criando um tipo tRestaurante, lendo o seu nome e, após isso, receber na entrada uma série de operadores que realizarão as seguintes operações:

**Operador 'A'**: cadastra um alimento num restaurante, isso é, armazena um tipo tAlimento na array de tAlimento de um restaurante, lendo a seguir as informações do alimento a partir da entrada.

**Operador 'C'**: Consome um alimento do restaurante, lendo a seguir o nome e a quantidade de alimento a ser consumida. Utilize o nome para pesquisar na array de alimentos de um restaurante (Dica: lembre-se das funções do .h que você implementou)

**Operador 'I'**: imprime as informações do restaurante, seguindo a formatação especificada

**Operador 'E'**: encerra o programa.

## Casos de Teste

Lembre-se que as entradas e saídas devem ser idênticas às dos casos de teste.

Exemplo de Entrada	Exemplo de Saida
	Nome: Ronaldo Lanches
	Alimentos:
	_____
A	Nome: Linguica de porco
Linguica de porco	Quantidade: 2000.00
26	Preco: R\$ 26.00
2000	Quantidade consumida: 0.00
A	Renda: R\$ 0.00
	_____
Arroz integral	_____
12	Nome: Arroz integral
10000	Quantidade: 9200.00
A	Preco: R\$ 12.00
Suco de maracujá	Quantidade consumida: 800.00
16	Renda: R\$ 9.60
1000	_____
C	_____
Arroz integral	Nome: Suco de maracujá
800	Quantidade: 1000.00
I	Preco: R\$ 16.00
E	Quantidade consumida: 0.00
	Renda: R\$ 0.00
	_____
	Quantidade total consumida: 800.00
	Renda total: R\$ 9.60