

Alunos:

Álvaro Ala Moraes Borges - 12121BSI269

Andressa Oliveira Bernardes – 12121BSI201

Giovanna Sara Lemos Guzzardi

## **TRABALHO PRÁTICO DE ESTRUTURA DE DADOS 1**

Após analisar todos os requisitos do trabalho foi feito uma elaboração de partes do processo:

Resumo do trabalho: TADs Fila e Pilha

- Cliente escolhe diferentes itens do cardápio;
- Itens disponíveis descritos em uma lista;
- Itens:
  - int codigo (identificação);
  - char refeicao (descrição, nome do item);
  - float preco (em real).

Na saída:

1. É feita uma fila de clientes onde os mesmos são atendidos pela gerente, cada cliente com uma comanda;
2. É informado para cada cliente atendido o preço total de sua comanda;
3. É dado um chocolate de tipo diverso para cada cliente, retirado de uma pilha.

Prática:

1. Preparar o cardápio
2. Preencher a fila de pessoas, cada uma com os produtos que consumiu
3. Preparar pilha com chocolates diversos
4. Mostrar ao final o preço total a ser pago por cada cliente e qual chocolate ele ganhou
5. Ter opção de imprimir a lista, fila ou pilha a qualquer momento
6. Ter os dados lidos de arquivos

De começo uma das primeiras coisas que foi percebida era a necessidade de entender como seriam armazenados o cardápio e brindes e para isso foram feitos dois arquivos que conteriam as opções fornecidas pelo restaurante. Essa primeira etapa foi de responsabilidade de Giovanna (12121BSI261).

1. Criação do arquivo:
  - a. Foi então criado o arquivo “cardápio.txt” onde seguimos os requisitos impostos (número de identificação (inteiro), a descrição (nome do item) e o preço (real));
  - b. No arquivo “chocolate.txt” havia apenas uma lista de doces que seriam a quantidade local de chocolate disponíveis.
2. Pegando os dados do arquivo:

- a. Uma das primeiras dificuldades foi entender como era feita a manipulação de arquivos no c, como fonte foi utilizado o livro LINGUAGEM C: COMPLETA E DESCOMPLICADA, do professor André Backes, para entendimento de como construir o código;
- b. Foi feita então criação de uma struct que de fato contivesse os itens do cardápio foi essência (struct item);
- c. Outra dificuldade foi entender como que os dados do cardápio seriam consumidos em outras partes do código, inicialmente a manipulação dos arquivos foi feita na main e os resultados foram mostrados com prints, mas houve uma quebra de raciocínio ao passar essa estrutura para uma função pois a passagem por referência deveria receber um vetor e retornar ele devidamente preenchido;
- d. A função "cardapio" teve uma complexidade menor devido se tratar de apenas uma leitura por referência que retornaria um vetor com todos os itens do cardápio;
- e. A função "brinde" é semelhante a função "cardapio", verifica se é possível abrir o arquivo chocolate.txt, e com o arquivo aberto e lido e empilhado cada chocolate da pilha de chocolates.

Após finalmente conseguir retornar os itens do cardápio e brinde foi então dividido a parte de criação da fila e pilha para a Andressa e Álvaro respectivamente.

Criação da Fila – Andressa (12121BSI201):

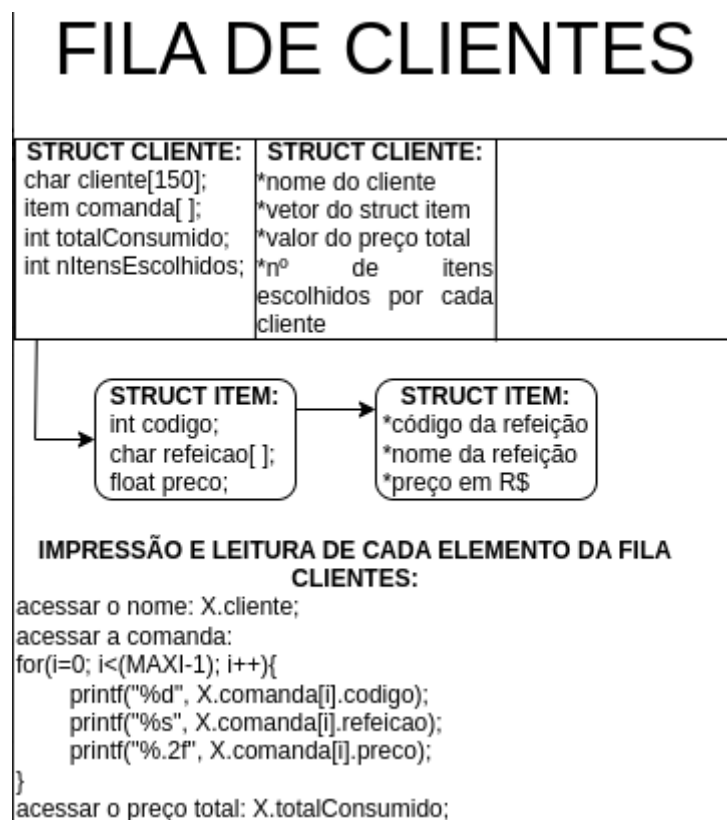
1. Logo após a Giovanna ter lidado com os arquivos, foi feita uma função auxiliar "apresentaMenu", que como o próprio nome diz, apresenta o menu para o cliente (imprime o menu na tela);
2. Em seguida, foi declarada a struct cliente, que continha os dados do cliente: nome (char), comanda (vetor de itens), total consumido (float, em reais) e o número de itens escolhidos por cada cliente;
3. Foram feitas todas as funções do TAD Fila para a filaClientes: criar, destruir, vazia, cheia, inserir e retirar, todas de forma semelhante as dos códigos de fila vistos em aula, porém com pequenas adequações;

**Obs.:** o TAD filaClientes foi alocada dinamicamente.

4. Para o melhor funcionamento das funções do TAD filaClientes, foram necessárias duas funções auxiliares:
  - a) a função "leItensCliente", que lê o nome do cliente, a quantidade de itens que o cliente deseja consumir (para não ficar um número fixo de itens escolhidos por cada cliente) e o código dos itens, que já seleciona, respectivamente, o nome da refeição e o preço da mesma... ela é usada para auxiliar a função insere;

- b) a função “imprimeElemento”, que imprime um elemento unitariamente, cada um dos campos da struct cliente. Ela auxilia a função retira, já que quando é retirado um elemento, ele é imprimido. Para melhor visualização de como ficaria a minha lista e como eu poderia acessar os elementos dela, fiz um diagrama no site Draw.io.

**Obs.:** o campo “totalConsumido” da struct cliente não é acessado é solicitado a leitura do código da refeição, pois é o compilador que faz esse cálculo, e o campo nItensEscolhidos não é acessado em nenhum momento, pois ele é apenas uma variável auxiliar.



5. Foram feitas também algumas funções auxiliares gerais:

- a) a função “total” recebe uma filaClientes e um elemento da mesma, soma todos os itens da comanda de cada cliente e devolve a mesma filaClientes, porém com o “totalConsumido” de cada cliente preenchido;
- b) a função “imprimeFilaClientes” imprime a filaClientes a qualquer momento; os itens são retirados da filaClientes FC passada por referência na função, imprimidos, inseridos na filaClientes auxiliar, retirados da auxiliar e inseridos novamente na FC.

6. Na main:

- a) é declarada e iniciada a filaClientes FC;
- b) é declarado um vetor de itens chamado menu e apresentado o menu para o cliente;
- c) é chamada a função inserir, juntamente com a sua auxiliar;
- d) é chamada a função "total" e logo em seguida, a função "imprimeFilaClientes";
- e) são chamadas algumas funções da pilha;
- f) por fim, a função é desenfileirada a filaClientes na função "entregaChocolates" e destruída no fim do código.

**Obs.:** a função "imprimeElemento" não é utilizada com a retirada na função "entregaChocolates" para não ter poluição visual na tela, caso contrário seria imprimido a mesma coisa duas vezes.

#### 7. Resultados:

- a) o resultado é lindo, assim que o código é compilado aparece o menu na tela e é solicitado o nome do cliente e a quantidade de itens que ele deseja, logo em seguida é solicitado o código dos itens desejados;
- b) em seguida, é impressa na tela a filaClientes completa, com o total a ser pago e por fim, qual chocolate cada cliente recebeu.

#### Criação da Pilha – Álvaro (12121BSI269)

1. Na terceira etapa, foi utilizada uma estrutura pilha juntamente com uma estrutura chocolates, para cumprir o requisito do trabalho de possuir uma pilha de chocolates diversos que estão descritos em um arquivo e são atribuídos a um cliente atendido que pagou sua comanda;
2. Primeiro a pilha é criada pela função criar;
3. Para ler chocolates do arquivo "chocolate.txt" utilizamos a função "brinde" (citada na parte da Giovanna).
4. Outras funções:
  - a) cada chocolate descrito no arquivo é empilhado na função brinde, e para isso ser feito utilizamos uma função chamada empilha;
  - b) a função empilha verifica se a pilha possui espaço e, caso sim, um chocolate é adicionado no topo da pilha;
  - c) os chocolates são empilhados na função "brinde";
  - d) a função desempilha, como o próprio nome diz, desempilha os chocolates da pilha;

- e) não foram utilizadas as funções vazia e cheia do TAD, mas foi feita essa verificação em cada função em que elas seriam utilizadas;
- f) para cada chocolate ser atribuído a um cliente, é necessário retirar o chocolate do topo da pilha, portanto é utilizado a função auxiliar “entregaChocolates”, que utiliza a desempilha (verificando se há chocolates na pilha), ao mesmo tempo que um cliente é tirado da filaClientes;
- g) por ultimo utilizamos a função “imprimePilhaChocolates” que desempilha os chocolates e empilham os mesmos em uma pilha auxiliar para exibir eles e depois retornam para a pilha original.

**Obs.:** foi utilizada alocação sequencial estática na pilha de chocolates.

5. Na main:

- a) usamos as funções citadas acima para realizar o objetivo do programa;
- b) por fim, utilizamos a função destruirPilha.

6. Resultados:

- a) depois dos resultados da filaClientes, é exibido na tela a pilha de chocolates antes de ser desempilhado qualquer elemento e o resultado da função “entregaChocolates” (qual cliente saiu e qual bombom ganhou).