

### **Especificação da quarta etapa:**

**Objetivo:** O objetivo desta quarta etapa é que o aluno aprenda a usar *funções, estruturas de dados do tipo vetor de registros, e estruturas de dados do tipo arquivo (binário e texto)* em algoritmos, e a implementá-las em programas.

**Problema:** CADASTRO DE PRATOS/BEBIDAS COM MENU DE OPÇÕES.

#### **Entrada/Saída:**

- Primeiramente deve ser mostrado ao usuário um menu de opções:

1. CRIAR E INICIAR CADASTRO NOVO
2. INCLUIR DADOS NO CADASTRO EXISTENTE
3. CONSULTAR O CADASTRO POR CODIGO DO PRATO
4. CONSULTAR O CADASTRO POR TIPO DE PRATO E FAIXA DE PREÇO
5. GERAR CARDÁPIOS EM ARQUIVO TXT
6. SAIR

- Se o usuário escolher a opção 1 (CRIAR E INICIAR CADASTRO NOVO): deve ser criado um arquivo binário sem dados, e devem ser lidos os seguintes dados para cada prato a ser armazenado no cadastro:

- Código
- Tipo ('E' para Entrada, 'P' para Prato Principal, 'S' para Sobremesa, e 'B' para Bebida)
- Descrição
- Preço

Após a leitura de cada prato, perguntar ao usuário se ele deseja incluir um novo prato (S/N). Se o usuário digitar 'N' retorne ao menu principal, caso contrário prossiga com a leitura de um novo prato.

Lembrando que caso o usuário selecione esta opção (1), se existir algum cadastro anterior este será totalmente apagado e recriado.

- Se o usuário escolher a opção 2 (INCLUIR DADOS NO CADASTRO EXISTENTE): nesta opção primeiramente deve ser testado se o arquivo binário já existe, caso contrário dar uma mensagem explicativa ao usuário (informando que o cadastro atual não possui dados) e automaticamente criar o arquivo (binário) novo vazio. Após deve-se prosseguir com o cadastro nesta opção mesmo.

Se o arquivo de dados já existir, então abrí-lo para leitura e adição de dados. Devem ser lidas e gravadas as mesmas informações do cadastro acima.

Após a leitura de cada prato, perguntar ao usuário se ele deseja incluir um novo prato (S/N). Se o usuário digitar 'N' retorne ao menu principal, caso contrário prossiga com a leitura de um novo prato.

- Se o usuário escolher a opção 3 (CONSULTAR O CADASTRO POR CODIGO DO PRATO): nesta opção primeiramente deve ser testado se o arquivo binário já existe. Se não existir, dar uma mensagem explicativa ao usuário e retornar ao menu principal. Caso contrário, o arquivo binário deve ser aberto para leitura. O usuário deve informar o código do prato que ele está buscando. Se o prato for encontrado no cadastro, exibir na tela todos os dados do prato. Caso contrário, exibir uma mensagem informando que o prato não existe no cadastro, e perguntar se o usuário deseja fazer uma nova busca (S/N). Se o usuário digitar 'N' retornar ao menu principal. Caso contrário, continuar lendo o código de outro prato para busca no cadastro.

obs: Pode-se assumir que não haverão pratos com códigos repetidos dentro do cadastro. Ou seja, você não precisa se preocupar com isso neste trabalho.

- Se o usuário escolher a opção 4 (CONSULTAR O CADASTRO POR POR TIPO DE PRATO E FAIXA DE PREÇO): nesta opção primeiramente deve ser testado se o arquivo já existe. Se não existir, dar uma mensagem explicativa ao usuário e retornar ao menu principal, pois não é possível realizar a consulta. Caso contrário, o arquivo binário deve ser aberto para leitura. O usuário deve informar o tipo de prato ('E', 'P', 'S', 'B') e a faixa de preço (limite mínimo e limite máximo) que ele deseja consultar. Considere 'E' para Entrada, 'P' para Prato Principal, 'S' para Sobremesa, e 'B' para Bebida.

**obs: Deve-se validar o tipo de prato ('E', 'P', 'S', 'B') informado e a faixa de preço, ou seja, o limite mínimo deve ser  $\geq 0$ , e o limite máximo deve ser  $\geq$  que o limite mínimo.**

Após, deve ser exibido na tela uma lista dos pratos cadastrados no tipo e na faixa de preço sendo consultados. Esta lista deve conter o código, descrição e preço de cada prato. **ATENÇÃO: esta lista deve estar em ordem alfabética de descrição do prato.** Logo após a lista de pratos, mostrar o número total de pratos listados (daquele tipo e faixa de preço consultados). No começo da listagem deve ser mostrado o número total de itens daquela listagem. Veja exemplo abaixo.

Caso não haja nenhum prato cadastrado naquele tipo e faixa de preço consultados, o programa deve mostrar uma mensagem indicando esta informação.

Após pergunta-se ao usuário se ele deseja prosseguir com a consulta por Tipo e Faixa de Preço (S/N). Se o usuário digitar 'N' retornar ao menu principal. Caso contrário, continuar lendo outro tipo e faixa de preço para busca no cadastro.

Caso a listagem seja muito grande e não caiba toda na tela, ela deve ser mostrada parcialmente com uma mensagem solicitando ao usuário teclar enter para prosseguir a visualização da listagem.

Exemplo da listagem (em ordem alfabética de DESCRIÇÃO) a ser mostrada na tela para a opção 4:

CONSULTA: PRATO PRINCIPAL  
TOTAL DE ITENS: 6

CODIGO	DESCRICAO	PREÇO
78	BACALHAU AO FORNO	31.00
31	ESPAGUETE AO MOLHO 4 QUEIJOS	29.00
19	FILE A PARMEGIANA	30.50
64	RISOTO DE CAMARAO	32.20
29	STROGONOFF DE CARNE	27.20
10	TILAPIA COM MOLHO DE ERVAS FINAS	29.90

- Se o usuário escolher a opção 5 (GERAR CARDÁPIOS EM ARQUIVO TXT) – deverá ser gerado um arquivo .txt contendo o cardápio mais barato e o cardápio mais caro levando em consideração todos os itens do cadastro. Nesta opção primeiramente deve ser testado se o arquivo binário que contém o cadastro já existe. Se não existir, dar uma mensagem explicativa ao usuário e retornar ao menu principal, pois não há dados para executar esta operação. Caso contrário, o arquivo binário deve ser aberto para leitura. Existem 2 possibilidades para a implementação desta opção:

- I. os registros do arquivo binário podem ser lidos e guardados em um vetor de registros, e depois de finalizar a leitura você percorre este vetor para encontrar e montar os cardápios mais caro e mais barato, ou
- II. pode-se ler o arquivo registro a registro e no momento da leitura já pode-se testar os dados do registro lido para montar o cardápio mais caro e mais barato.

**Sugestão:** Tanto na opção I como na opção II, ao invés de você ter uma série de variáveis para guardar cada item mais caro e mais barato (como foi feito no trabalho 1), você pode declarar um vetor de registros para ir guardando os registros dos itens mais caros e mais baratos, e testar esses registros mesmo. Para isso, basta um vetor de registros de 8 posições, já que são 2 cardápios, cada um com 4 itens. Basta você determinar, por exemplo, que os 4 primeiros registros guardam os itens do cardápio mais barato e os 4 últimos guardam os itens do cardápio mais caro. Não se esqueça que caso você faça, já na primeira vez, o teste do campo preço do vetor de registros é necessário inicializar esse campo com os valores adequados.

Após determinar o cardápio mais caro e mais barato, os dados devem ser gravados em disco num arquivo tipo texto com o nome “CARDAPIOS.txt”. Ver abaixo exemplo de saída a ser gravado no arquivo .txt. Lembre-se de que a gravação deste arquivo deve ser feita linha a linha.

Nesta opção o usuário não verá nada na tela (apenas quando abrir o arquivo CARDAPIOS.txt em um editor de textos). Portanto, após gerar o arquivo texto exibir uma mensagem explicativa: Arquivo texto gerado com sucesso! Tecle <enter> para continuar...

Exemplo do conteúdo do arquivo “CARDAPIOS.TXT”:

```
*****
*** CARDAPIO MAIS BARATO ***
*****

*** ENTRADA ***
CODIGO: 13
DESCRICAO: CREME DE ASPARGOS
PRECO: 10.50

*** PRATO PRINCIPAL ***
CODIGO: 29
DESCRICAO: STROGONOFF DE CARNE
PRECO: 27.20

*** SOBREMESA ***
CODIGO: 63
DESCRICAO: SORVETE COM CALDA DE CHOCOLATE
```

PRECO: 8.10

\*\*\* BEBIDA \*\*\*

CODIGO: 66

DESCRICAO: AGUA MINERAL

PRECO: 3.50

VALOR TOTAL DO CARDAPIO: 49,30

\*\*\*\*\*

\*\*\* CARDAPIO MAIS CARO \*\*\*

\*\*\*\*\*

\*\*\* ENTRADA \*\*\*

CODIGO: 76

DESCRICAO: COQUETEL DE CAMAROEES

PRECO: 18.20

\*\*\* PRATO PRINCIPAL \*\*\*

CODIGO: 64

DESCRICAO: RISOTO DE CAMARAO

PRECO: 32.20

\*\*\* SOBREMESA \*\*\*

CODIGO: 59

DESCRICAO: PETIT GATEAU

PRECO: 13.80

\*\*\* BEBIDA \*\*\*

CODIGO: 60

DESCRICAO: TACA DE VINHO TINTO

PRECO: 10.00

VALOR TOTAL DO CARDAPIO: 74.20

- Se o usuário escolher a opção 6 (SAIR): encerrar o programa.

### **Detalhamento do problema:**

1. As unidades de entrada e saída a serem utilizadas serão disco ('RESTAURANTE.bin') e monitor de vídeo e arquivo ('CARDAPIOS.txt'), respectivamente.

Obs: É importante observar que neste trabalho você não sabe quantos pratos estarão cadastrados no arquivo. O usuário poderá sair de uma opção do menu e entrar em outra, e posteriormente retornar e continuar o cadastro. É possível ainda que o usuário saia do programa e entre novamente para cadastrar mais pratos no mesmo arquivo. Assim, não é mantido um contador da quantia de pratos gravados no arquivo.

2. Para facilitar o manuseio do programa, deverão ser dadas mensagens solicitando os dados de entrada, e mensagens explicativas das informações de saída fornecidas pelo programa.

3. A primeira ação do programa deve ser a apresentação do MENU PRINCIPAL:

1. CRIAR E INICIAR CADASTRO NOVO
2. INCLUIR DADOS NO CADASTRO EXISTENTE
3. CONSULTAR O CADASTRO POR CODIGO DO PRATO
4. CONSULTAR O CADASTRO POR TIPO DE PRATO E FAIXA DE PREÇO
5. GERAR CARDÁPIOS EM ARQUIVO TXT
6. SAIR

4. Tipo dos dados:

- o código deve ser numérico do tipo int.
- o tipo do prato deve ser um caracter (char)
- a descrição do prato/bebida deve ser char[41]
- o preço deve ser numérico do tipo float.

5. Testes de consistência que deve ser feitos durante a leitura dos dados:

OBS: Utilizar a estrutura de repetição “**while**” (ou “**do-while**”) para testar a consistência e continuar lendo o campo caso um valor inválido tenha sido inserido pelo usuário. Fornecer mensagens explicativas quando ocorrer um erro na entrada de dados.

- o código e preço devem ser maior do que zero (>0).
- o tipo do prato deve ser igual a ‘E’, ‘P’, ‘S’, ou ‘B’.
- a resposta para a pergunta de continuidade do cadastro, deve ser somente ‘S’, ‘s’, ‘N’, ou ‘n’.

Para os demais campos (abaixo), **NÃO é necessário** testar a consistência, ou seja, pode-se assumir que o usuário irá informar o conteúdo corretamente.

- descricao

6. **IMPORTANTE:** Veja no slide nro 12 do conteúdo sobre STRINGS (no moodle) como permitir que o comando *scanf* leia strings que contenham espaços em branco.

7. Pode-se assumir que não haverá itens (pratos/bebidas) com preços iguais.

8. Use variáveis **Mnemônicas** (significativas) no seu programa fonte.

9. Nas opções 1 e 2 do MENU PRINCIPAL, referentes a gravação dos registros no arquivo, você pode implementar de duas formas:

I. Pode-se ler um registro do teclado e gravá-lo imediatamente no arquivo, ou

II. Pode-se ler vários registros do teclado e colocar em um vetor de registros e gravá-los em bloco, ou seja, gravar vários registros do vetor de uma só vez. Lembre-se, no entanto, de que algumas questões devem ser observadas neste caso: você não sabe quantos itens o usuário vai cadastrar, portanto, a sugestão é que você determine um tamanho para o vetor de registros, 10 por exemplo, e cada vez que o usuário preencher o vetor, você descarrega ele no disco, ou seja, grava os 10 registros no arquivo binário. O

usuário não tem conhecimento dessa solução, ou seja, ele imagina que está cadastrando e gravando, quando na verdade você guarda na RAM e quanto completar 10 itens você efetua a gravação propriamente dita. É fundamental que esta solução com vetor de registros também funcione para a situação em que o usuário cadastrar menos de 10 registros. Neste caso, você deve gravar somente o número de registros cadastrados e não o vetor inteiro, pois as posições finais estarão vazias e não devem ir para o arquivo.

**ATENÇÃO:** Em hipótese alguma você deve guardar ou tentar guardar em memória RAM o número de registros que existem dentro do arquivo. O arquivo constantemente alterado e a solução deve funcionar para um número indeterminado de informações gravadas nele. Lembre-se, o limite do tamanho do arquivo é o HD (disco rígido).

**10. Para cada opção do menu principal deverá ser chamado uma FUNÇÃO com passagem de parâmetro(s):**

- para todas as funções correspondentes às opções do menu arquivo de dados deve ser passada uma variável do tipo string contendo o nome do arquivo físico onde os dados serão gravados (“RESTAURANTE.bin”, por exemplo).
- para a função que vai gerar o arquivo tipo texto, deverão ser passados dois parâmetros, ou seja, duas strings contendo o nome do arquivo físico binário e do arquivo físico do tipo texto. (“CARDAPIOS.txt”, por exemplo).
- para a opção 1 e 2 deve ser chamada a mesma função. Passar como parâmetro uma variável contendo o número da opção escolhida. Com esta informação é possível saber se o cadastro deve ser recriado ou apenas aberto para inclusão de novos dados.

Dentro de cada função deverão ser definidos os ponteiros para os arquivos, e estes deverão ser abertos utilizando os modos de aberturas correspondentes para cada situação, e fechados corretamente.

11. No item 4 do MENU PRINCIPAL será necessário ordenar, alfabeticamente por descrição do prato, os dados a serem mostrados na tela. Neste caso, crie uma estrutura do tipo vetor de registros, jogue os registros que serão manipulados para esse vetor (um a um), e ordene o vetor (**não o arquivo!**). Depois mostre os dados desse vetor (registro a registro) na tela.

12. Para saber se o arquivo binário chegou no final, basta testar se o número de bytes lidos com a função *fread* é igual a zero. (Veja isso nos slides.)

13. Caso necessário, pode-se criar funções além das mencionados acima.

14. Para implementar o menu de opções é permitido utilizar a estrutura *switch ... case* do C (consultar apostila disponibilizada no moodle), ou então *if*.

**Observações Gerais:**

1. Incluir cabeçalho como comentário (ou seja, entre /\* \*/), no programa fonte, de acordo com os critérios de avaliação dos trabalhos (Disponível no Moodle).
2. A data de entrega do programa é: **31/01/2011 (2ª-feira) até às 23:55 hs.**