

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 IMOBILIÁRIO (VENDA/ALUGUEL) 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 DE IMOVEL
4. CONSULTAR O CADASTRO POR CATEGORIA E TIPO DE IMOVEL
5. GERAR ARQUIVO TXT DO CADASTRO DE VENDA
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 imóvel a ser armazenado no cadastro:

- Código do imóvel
- Categoria (“V” para venda, ou “A” para aluguel)
- Tipo (1-apartamento, 2-casa, 3-lote, 4-loja)
- Endereço
- Número de quartos (digitar 0 para kitinete, lote, e loja)
- Garagem (S/N)
- Metragem
- Preço
- Observações
- Data de cadastro

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

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

- Se o usuário escolher a opção 3 (CONSULTAR O CADASTRO POR CODIGO DO IMOVEL): 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 imóvel que ele está buscando. Se o imóvel for encontrado no cadastro, exibir na tela todos os dados do imóvel. Caso contrário, exibir uma mensagem informando que o imóvel 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 imóvel para busca no cadastro.

obs: Pode-se assumir que não haverão imóveis 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 CATEGORIA E TIPO DE IMÓVEL): 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 a categoria e o tipo de imóvel que ele deseja consultar.

obs: Deve-se validar a categoria ('A' ou 'V') e o tipo (1, 2, 3 ou 4) informados.

Após, deve ser exibido na tela uma lista dos imóveis cadastrados na categoria e tipo sendo consultados. Esta lista deve conter o código, endereço, e preço de cada imóvel. **ATENÇÃO: esta lista de imóveis deve estar em ordem decrescente de preço.** Logo após a lista de imóveis, mostrar o número total de imóveis listados (naquela categoria e tipo consultados).

Caso não haja nenhum imóvel cadastrado naquela categoria e tipo, o programa deve mostrar uma mensagem indicando esta informação.

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

Se o usuário escolher a opção 5 (GERAR ARQUIVO TXT DO CADASTRO DE VENDA - em ordem **crescente** de tipo e em ordem **decrescente** de preço dentro de cada sublista de tipo): 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 há dados para executar esta operação. Caso contrário, o arquivo binário deve ser aberto para leitura. Os dados do arquivo binário devem ser lidos e guardados em um vetor de registros (somente imóveis a VENDA). Este vetor deve ser primeiramente classificado por ordem crescente de tipo, e posteriormente em ordem decrescente de preço para cada sublista de tipo dentro do vetor. Após os dados devem ser gravados em disco num arquivo tipo texto com o nome "IMOVEIS_VENDA.txt". Em cada linha do arquivo texto devem estar gravados os dados (tipo, código, endereço e preço) de um IMOVEL a venda. (Ver exemplo abaixo.) Caso não tenha sido cadastrado nenhum imóvel para uma determinado tipo, este tipo não deve aparecer no relatório gravado no arquivo txt.

Nesta opção o usuário não verá nada na tela (apenas quando abrir o arquivo IMOVEIS_VENDA.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...

(Os dados abaixo são fictícios, e podem não representar um situação real)

*** LISTAGEM DE IMOVEIS PARA VENDA ***

TIPO	CODIGO	ENDERECO	PRECO
1	981	SQS 202 bloco A apto 302	800000.00
1	655	SQN 105 bloco F apto 401	760000.00
1	504	SQSW 300 bloco L apto 106	743000.00
1	561	SQN 415 bloco B apto 101	550000.00
2	114	SHIS QL 12 cj 3 casa 5	990000.00
2	699	SHIN QI 5 cj 2 casa 18	880000.00
2	230	SHIN QL 12 cj 8 casa 9	790000.00
3	881	SHIS QI 25 cj 4 lote 45	830000.00
3	117	SHTQ quadra 3 lote 2	270000.00
4	229	SQN 315 bloco D	230000.00

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 ('IMOVEIS.bin') e monitor de vídeo e arquivo ('IMOVEIS_VENDA.txt'), respectivamente.

Obs: Neste trabalho não haverá um contador único da quantia de imóveis cadastrados como foi feito no trabalho de vetores. Na verdade, você não sabe quantos imóveis 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 imóveis no mesmo arquivo. Assim, não é mantido um contador da quantia de imóveis 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 DE IMOVEL
4. CONSULTAR O CADASTRO POR CATEGORIA E TIPO DE IMOVEL
5. GERAR ARQUIVO TXT DO CADASTRO DE VENDA
6. SAIR

4. Tipo dos dados:

- categoria deve ser do tipo char, o campo Endereço deve ser char[41], o campo Garagem deve ser do tipo char, o campo observações deve ser do tipo char[51], e a data de cadastro deve ser do tipo char[9] (formato DD/MM/AA),

- o código, o tipo, o numero de quartos devem ser numéricos do tipo int.

- a metragem e o preço deve ser numéricos 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.

- **Código:** não é permitido entrar com zero ou números negativos. Se o usuário assim o programa deve dar uma mensagem de erro, e continuar lendo esta informação até que seja informado um dado válido.

- **Categoria:** deve ser somente “V” para venda, ou “A” para aluguel. Fazer teste de consistência conforme explicado acima para código.

- **Tipo:** deve ser somente 1, 2, 3, ou 4, sendo que 1 significa apartamento, 2 significa casa, 3 significa lote, e 4 significa loja. Fazer teste de consistência conforme explicado acima para código.

- **Endereço:** não é permitido deixar este campo em branco. Fazer teste de consistência conforme explicado acima para código.

- **Número de quartos:** deve ser um valor positivo ou zero para kitinete, lote, e loja). Quando o usuário digitar 3 ou 4 no campo tipo, aqui o valor deve obrigatoriamente ser zero. Fazer teste de consistência conforme explicado acima para código.

- **Garagem:** deve-ser somente ‘S’, ‘s’, ‘N’, ou ‘n’. Fazer teste de consistência conforme explicado acima para código.

- **Metragem:** deve ser um valor positivo. Fazer teste de consistência conforme explicado acima para código.

- **Preço:** deve ser um valor positivo. Fazer teste de consistência conforme explicado acima para código.

- **A resposta** para a pergunta de continuidade do cadastro, deve-ser somente ‘S’, ‘s’, ‘N’, ou ‘n’. Fazer teste de consistência conforme explicado acima para código.

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

- Observações,
- Data do cadastro

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 haverão imóveis com código, preços e/ou metragem iguais.

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

9. Você deve definir um registro, e um arquivo de registros para armazenar e manipular as informações do cadastro. Utilize um **vetor de registros somente** para trabalhar com a ordenação de dados, que serão posteriormente gravados no arquivo tipo texto.

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 (“IMOVEIS.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. (“IMOVEIS_VENDA.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.

11. Sempre que precisar ordenar os registros do arquivo de dados (binário), 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 grave os dados desse vetor (registro a registro) no arquivo tipo texto (“IMOVEIS_VENDA.txt”), ou mostre na tela, conforme a necessidade da solução.

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.

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*.

15. A entrada de dados deste trabalho é similar a entrada de dados do trabalho 2 (sobre vetores), com a principal diferença que neste trabalho você irá utilizar um registro para ler os dados e gravá-lo no arquivo binário, e não um vetor como no trabalho 2. Caso deseje, você pode utilizar parte do código do trabalho 2 como auxílio para escrever o código deste trabalho.

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: **15/12/2009 (3ª-feira) até às 23:55 hs. Haverá uma tolerância até às 8 hs da manhã do dia seguinte.**

3. Como este é o último trabalho da disciplina e existe um prazo para entrega das menções **serão aceitos trabalhos SOMENTE com no máximo 2 dias de atraso (efetuando os devidos descontos).** Após esse prazo não serão aceitos trabalhos.