

Trabalho Prático 2

Objetivo: O objetivo desta 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 JOGOS DE FUTEBOL 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 NÚMERO DO JOGO
4. CONSULTAR O CADASTRO POR RODADA
5. GERAR RELATÓRIO 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 jogo a ser armazenado no cadastro:

- número do jogo
- nome do time da casa
- nome do time visitante
- estádio do jogo
- data do jogo
- hora do jogo
- rodada do campeonato
- público (número de pagantes)
- renda do jogo (bilheteria)
- número de gols do time da casa
- número de gols do time visitante

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

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 mesma opção.

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

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

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

Obs: Pode-se assumir que não haverá jogos com números 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 RODADA): 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 rodada que deseja consultar. Nesse momento, **deve-se validar a rodada, ou seja, deve ser > 0.**

Após, deve ser exibido na tela uma lista dos jogos cadastrados na rodada sendo consultada. Esta lista deve conter o número do jogo, o time da casa, o time visitante e a data do jogo. **ATENÇÃO: esta lista deve estar em ordem crescente de número do jogo.** No começo da listagem deve ser mostrado a rodada consultada e o número total de jogos daquela listagem. Veja exemplo abaixo.

Caso não haja nenhum jogo cadastrado naquela rodada, o programa deve mostrar uma mensagem indicando esta informação.

Após pergunta-se ao usuário se ele deseja prosseguir com a consulta por Rodada (S/N). Se o usuário digitar 'N' retornar ao menu principal. Caso contrário, continuar lendo outra rodada 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 NÚMERO DO JOGO) a ser mostrada na tela para a opção 4:

RODADA: 3

TOTAL DE JOGOS: 4

NÚMERO	TIME CASA	TIME VISITANTE	DATA
18	CORITIBA	BOTAFOGO	10/07/12
19	GREMIO	FLAMENGO	11/07/12
24	CORINTHIANS	BAHIA	11/07/12
30	PALMEIRAS	FLUMINENSE	12/07/12

- Se o usuário escolher a opção 5 (GERAR RELATÓRIO EM ARQUIVO TXT) – deverá ser gerado um arquivo de texto “.txt” contendo um relatório de todos os jogos do cadastro agrupados por rodada (em ordem crescente de rodada). 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 e os registros do arquivo devem ser lidos e guardados em um vetor de registros. Após finalizar a leitura você deve percorrer este vetor para processar e gerar o relatório que será gravado no arquivo tipo texto com o nome “RELATORIO.txt”.

ATENÇÃO: O arquivo texto não deverá conter as mensagens solicitando ao usuário pressionar <Enter> entre os dados de cada rodada. 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 RELATORIO.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 “RELATORIO.TXT”:

```
*****
*** RODADA: 1a ***
*****
```

```
JOGO (CASA X VISITANTE): AVAI X SAO PAULO
ESTADIO: RESSACADA
DATA: 16/03/12          HORA: 20:00
PUBLICO: 42540          RENDA (R$): 756400.00
RESULTADO (CASA X VISITANTE): 2 X 1
```

```
JOGO (CASA X VISITANTE): FLUMINENSE X VASCO
ESTADIO: LARANJEIRAS
DATA: 17/03/12          HORA: 16:30
PUBLICO: 29833          RENDA (R$): 549211.00
RESULTADO (CASA X VISITANTE): 2 X 2
```

```
JOGO (CASA X VISITANTE): CRUZEIRO X INTERNACIONAL
ESTADIO: MINEIRAO
DATA: 18/03/12          HORA: 16:30
PUBLICO: 37930          RENDA (R$): 784421.00
RESULTADO (CASA X VISITANTE): 5 X 0
```

```
NUMERO TOTAL DE JOGOS (DA RODADA): 3
MEDIA DE GOLS POR JOGO (DA RODADA): 4
```

```
*****
*** RODADA: 2a ***
```

JOGO (CASA X VISITANTE): GREMIO X SANTOS
ESTADIO: OLIMPICO
DATA: 02/04/12 HORA: 16:00
PUBLICO: 32590 RENDA (R\$): 860455.00
RESULTADO (CASA X VISITANTE): 2 X 0

JOGO (CASA X VISITANTE): FLAMENGO X CEARA
ESTADIO: MARACANA
DATA: 04/04/12 HORA: 18:00
PUBLICO: 54377 RENDA (R\$): 1241890.00
RESULTADO (CASA X VISITANTE): 0 X 3

NUMERO TOTAL DE JOGOS (DA RODADA): 2
MEDIA DE GOLS POR JOGO (DA RODADA): 2.5

*** RODADA: 3a ***

JOGO (CASA X VISITANTE): CORITIBA X BOTAFOGO
ESTADIO: COUTO PEREIRA
DATA: 10/04/12 HORA: 16:00
PUBLICO: 21983 RENDA (R\$): 431000.00
RESULTADO (CASA X VISITANTE): 3 X 1

JOGO (CASA X VISITANTE): GREMIO X FLAMENGO
ESTADIO: OLIMPICO
DATA: 11/04/12 HORA: 16:30
PUBLICO: 37930 RENDA (R\$): 678940.00
RESULTADO (CASA X VISITANTE): 1 X 0

JOGO (CASA X VISITANTE): CORINTHIANS X BAHIA
ESTADIO: PARQUE SAO JORGE
DATA: 11/04/12 HORA: 16:00
PUBLICO: 31943 RENDA (R\$): 753022.00
RESULTADO (CASA X VISITANTE): 3 X 1

JOGO (CASA X VISITANTE): PALMEIRAS X FLUMINENSE
ESTADIO: PALESTRA ITALIA
DATA: 12/04/12 HORA: 16:00
PUBLICO: 41422 RENDA (R\$): 1105803.00
RESULTADO (CASA X VISITANTE): 1 X 1

NUMERO TOTAL DE JOGOS (DA RODADA): 4
MEDIA DE GOLS POR JOGO (DA RODADA): 2.75

*** RODADA: 5a ***

JOGO (CASA X VISITANTE): SANTOS X CRUZEIRO
ESTADIO: VILA BELMIRO
DATA: 02/04/12 HORA: 17:00
PUBLICO: 29860 RENDA (R\$): 620100.00
RESULTADO (CASA X VISITANTE): 3 X 3

JOGO (CASA X VISITANTE): INTERNACIONAL X VASCO
ESTADIO: BEIRA-RIO
DATA: 04/04/12 HORA: 20:00
PUBLICO: 27408 RENDA (R\$): 543900.00
RESULTADO (CASA X VISITANTE): 1 X 2

NUMERO TOTAL DE JOGOS (DA RODADA): 2
MEDIA DE GOLS POR JOGO (DA RODADA): 4.5

*** DADOS GERAIS DO CAMPEONATO ***

NUMERO TOTAL DE JOGOS: 11
MEDIA DE GOLS POR JOGO: 3.36
MEDIA DE PUBLICO POR JOGO: 35256
MEDIA DE RENDA POR JOGO: 756831.09
NUMERO DE VITORIAS DO TIME DA CASA: 6
NUMERO DE VITORIAS DO TIME VISITANTE: 2
NUMERO DE EMPATES: 3
JOGO COM MAIOR RENDA:
FLAMENGO (0) X CEARA (3), MARACANA, DIA 04/04/12, 2a RODADA,
PUBLICO 54377, RENDA R\$ 1432466.00

- 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 ('JOGOS.bin') e monitor de vídeo e arquivo ('RELATORIO.txt'), respectivamente.

Obs: É importante observar que neste trabalho você não sabe quantos jogos 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 jogos no mesmo arquivo. **Assim, não é mantido um contador da quantia de jogos 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 NÚMERO DO JOGO
4. CONSULTAR O CADASTRO POR RODADA
5. GERAR RELATORIO EM ARQUIVO TXT
6. SAIR

4. Tipo dos dados:

- número do jogo deve ser do tipo int
- nome do time da casa deve ser char[16]
- nome do time visitante deve ser char[16]
- estádio do jogo deve ser char[21]
- data do jogo deve ser char[9]
- hora do jogo deve ser char[6]
- rodada do campeonato deve ser numérico do tipo int.
- público (número de pagantes) deve ser numérico do tipo int.
- renda do jogo (bilheteria) deve ser numérico do tipo float.
- número de gols do time da casa deve ser numérico do tipo int.
- número de gols do time visitante deve ser numérico do tipo int.

5. Testes de consistência que devem 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 número do jogo, rodada do campeonato, público (número de pagantes), renda do jogo (bilheteria) devem ser maior do que zero (>0).
- número de gols do time da casa e número de gols do time visitante devem ser maior ou igual a zero (>=0).
- a resposta para a pergunta de continuidade do cadastro, deve ser somente ‘S’, ‘s’, ‘N’, ou ‘n’.
- nome do time da casa, nome do time visitante, nome do estádio do jogo e hora do jogo não podem ser deixados em branco, ou seja, o usuário não pode simplesmente apertar <Enter> nem digitar apenas espaços em branco.

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

- Data do cadastro

6. **IMPORTANTE:** Veja no slides 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 jogos com números iguais no cadastro.

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 quando 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 chamada 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 (“JOGOS.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. (“RELATORIO.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, em ordem crescente de número do jogo, os dados a serem mostrados na tela. Neste caso, crie uma estrutura do tipo vetor de registros, jogue os registros que serão manipulados (no caso os registros da rodada consultada)

para esse vetor (um a um), e ordene o vetor (**não o arquivo!**). Depois mostre os dados (somente os necessários) desse vetor (registro a registro) na tela. **OBS: A ordenação de um vetor de registros é simples, já que você pode trocar um registro inteiro de posição dentro do vetor, sem a necessidade de copiar cada campo do registro.**

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: **06/07/2016 (4ª-feira) até às 23:55 hs.**