

Trabalho Prático 3

O objetivo desta terceira etapa é que o aluno aprenda a usar **estruturas de dados do tipo matriz, subalgoritmos (funções) e manipulação de arquivos** em algoritmos, bem como implementá-los corretamente em programas.

Especificação da terceira etapa:

Problema: Implementar um jogo similar ao jogo “Tetris”, que vamos chamar de “CBestris”, ou seja, o Tetris da Disciplina de CB.

Detalhamento do problema:

1. As unidades de entrada deste trabalho deverão ser o disco rígido (um arquivo contendo todas as sequências do jogo), bem como o teclado para inserção dos comandos de manipulação das sequências. A unidade de saída será o monitor de vídeo. O arquivo de entrada será padrão e fornecido pela professora. Detalhes sobre a leitura do arquivo serão dados abaixo.
2. Para facilitar o manuseio do programa, deverão ser dadas mensagens explicativas em alguns casos conforme indicado posteriormente nesta especificação.
3. Inicialmente deve ser mostrada ao usuário a seguinte tela inicial, até que o usuário digite <enter>.

```

*** CBetris 2009-2 ***
+-----+
|
|
|
|
|
|=>+
|
|
|
|
|
+-----+
+-----+

```

Tecle <enter> para iniciar o jogo

4. A área do jogo deverá ser dividida em duas partes, uma superior e outra inferior. Estas áreas são demarcadas pelas setas nas laterais (\Rightarrow \Leftarrow). A área superior é reservada para a apresentação e manipulação das sequências do jogo. A área inferior é reservada para o posicionamento das sequências após a manipulação pelo jogador.
5. **Leitura do arquivo contendo as sequências do jogo:** será fornecido um arquivo contendo as sequências (unidimensionais) que o jogo irá manipular, denominado “sequencias.txt”. Este arquivo, do tipo texto, deverá ser aberto, lido completamente e armazenado em uma matriz na memória RAM. Após a leitura, o arquivo deve ser fechado, e o programa terá acesso somente a matriz contendo as sequências. Cada sequência (considerada uma string no arquivo), deve ser armazenada em uma linha da matriz.
6. **Manipulação das sequências:** o jogo inicia com uma sequência sendo apresentada (sempre) VERTICALMENTE na área superior do jogo. O usuário tem as seguintes opções:
R-rotar D-direita E-esquerda C-cair (ver ilustração abaixo)

```
*** CBetris 2009-2 ***  
+-----+ PONTUACAO: 0  
|          * |  
|          * |  
|          * |  
|          * |  
|          * |  
|          | |  
|= > +      + < = |  
|          | |  
|          | |  
|          | |  
|          | |  
|          | |  
+-----+
```

R-rotar D-direita E-esquerda C-cair
ACAO:

O seu programa deve ler e executar a ação digitada pelo usuário. O programa deve aceitar somente as letras R, D, E ou C (tanto maiúscula como minúscula). Não é necessário dar mensagem de erro, mas o programa não deve executar nada enquanto o usuário não digitar uma letra correta.

```

*** CBetris 2009-2 ***
+-----+
|                                     |
|               *****             |
|                                     |
|                                     |
|                                     |
| =>+                                     | +<=
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
+-----+
R-rotar D-direita E-esquerda C-cair
ACAO: r

```

```

*** CBetris 2009-2 ***
+-----+
|                                     | # |
|                                     | # |
|                                     | # |
|                                     | # |
|                                     |   |
|                                     |   |
|=>+                               +<=|
|                                     |   |
|                                     |   |
|                                     |   |
|                                     |   |
|                                     |   |
|                                     |   |
| $$$                             *****|
+-----+
R-rotar D-direita E-esquerda C-cair
ACAO:
```

E-esquerda: move a sequência em apenas 1 posição para a esquerda da tela. A sequência não pode ultrapassar a área limite do jogo. Nesta opção a sequência deve ser movida no mesmo sentido em que se encontra (vertical ou horizontal).

```

*** CBetris 2009-2 ***
+-----+
|
|$$$
|
|
|
|
|=>+
|
|
|
|
|
|
|
|
|
|
+-----+
|
|*****
|
+-----+
PONTUACAO: 0
R-rotar D-direita E-esquerda C-cair
ACA0: e

```

Obs: o usuário pode ficar digitando as 3 opções acima quantas vezes quiser. O loop que controla essas ações só é encerrado quando o usuário digitar “C” (cair).

C-cair: esta opção remove a sequência da área superior do jogo para a área inferior do jogo. A sequência deve ser posicionada nas mesmas colunas em que se encontra, porém na primeira linha totalmente disponível na área inferior, de modo que a sequência não seja quebrada, e também não sobreponha nenhuma outra sequência existente. (ver a ilustração abaixo)

```
*** CBetris 2009-2 ***  
+-----+ PONTUACAO: 0  
|           #           |  
|           #           |  
|                   |  
|                   |  
|                   |  
|=>+                   +<=  
|                   |  
|                   |  
|                   |  
|           #####      |  
|           *          $   |  
|       *    *          $   |  
|  $$$*    *     #####*****|  
+-----+  
R-rotar D-direita E-esquerda C-cair  
ACAO:
```

- ```
PONTUACAO: 0
```
- ```
+-----+
|               |
|               |
|               |
|               |
|               |
|=>+           +<=
|               |
|#              |
|#              |
|#              |
|# #            |
|# # $ * **#    |
|XXXXXXXXXXXXXXXXXXXXX|
+-----+
```
- R-rotar D-direita E-esquerda C-cair

8. Pontuação: no lado direito da área do jogo deve ser mostrada a pontuação, que inicia com 0(zero) e cada linha que o jogador consegue preencher e eliminar na parte inferior

do jogo, vale 100 pontos. Os pontos devem ser verificados e incrementados imediatamente após a eliminação da(s) linha(s).

9. O jogo termina em 2 situações:

1-quando o jogador conseguir chegar ao fim de todas as sequências lidas do arquivo e tiver atingido a altura limite da área inferior, delimitada pelas setas => <=. Quando o jogador chegar ao fim das sequências deve ser mostrada a mensagem abaixo juntamente com a pontuação:

```
*** PARABENS!!! VOCE CHEGOU AO FIM DO JOGO COM 800 PONTOS!!!
Tecle <enter> para sair
```

2-quando as sequências posicionadas na parte inferior da tela atingirem a altura limite (indicada pelas setas => <=), a matriz que armazena o jogo deve ser preenchida com “X” e mostrada na tela juntamente com a mensagem “You Lose!!! Tecle <enter> para sair.” Veja ilustração abaixo.

```
ACAO:      *** CBetris 2009-2 ***
+-----+
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
=>+XXXXXXXXXXXXXXXXXXXXX+<=
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
|XXXXXXXXXXXXXXXXXXXXX|
+-----+
R-rotar D-direita E-esquerda C-cair
ACAO:
*** YOU LOSE!!!Tecle <enter> para sair
```

10. ATENÇÃO: após cada atualização, você deve limpar a tela e mostrar a matriz na mesma posição anterior. Isso dará ao usuário a impressão de que as sequências estão em movimento ou realmente “caindo”.

11. Neste trabalho você DEVE modularizar o seu programa criando funções. Conforme já dito em sala de aula, funções tornam o seu programa mais legível, mais fácil de se entender, e mais curto, já que evita a duplicação de código para uma mesma ação.

*** Deverão ser criadas, no mínimo, as seguintes funções:**

- 1 - Função para mostrar a tela inicial (ainda sem execução do jogo);
- 2 - Função para limpar a tela e mostrar a matriz (e pontos) na tela;
- 3 - Função para ler e identificar as 4 ações (R-rotar D-direita E-esquerda C-cair);

- 4 - Função para Rotacionar (e alterar a matriz) – esta função será chamada pela função acima (das ações)
- 5 - Função para Mover para mover a sequência para direita (e alterar a matriz) – esta função será chamada pela função acima (das ações)
- 6- Função para Mover a sequência para esquerda (e alterar a matriz) – esta função será chamada pela função acima (das ações)
- 7-Função para “cair a sequência”, ou seja mover da parte superior para a parte inferior do jogo (e alterar a matriz) – esta função será chamada pela função acima (das ações)
- 8-Função para verificar as linhas que foram preenchidas e devem ser eliminadas (esta função também altera a matriz eliminando as linhas cheias).
- 9-Função para ler as sequências do arquivo e guardar em uma matriz.
- 10- Função para inserir uma nova sequência na parte superior da matriz.
- 11- Função para preencher toda a matriz do jogo com “X” quando jogador perder o jogo.

Você pode criar outras funções (não obrigatórias) caso ache necessário!

- 12. Crie variáveis e nomes de funções mnemônicos!!!
- 13. NÃO utilize variáveis globais!
- 14. Você trabalhará com 2 matrizes: uma para guardar as sequências que foram lidas do arquivo e serão inseridas no jogo, e outra para guardar o jogo propriamente dito, ou seja, armazene tanto a parte inferior como a parte superior do jogo em uma matriz só. A parte da moldura (frame) não deve ser armazenada junto na matriz. Coloque os caracteres da moldura diretamente no comando printf.
- 15. Nas funções, utilize **passagem de parâmetros por valor e por referência**, conforme necessário.
- 16. Faça diversos testes no seu jogo, jogando é claro. Bom trabalho e boa diversão!

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: **06/12/2009 (Dom.) até às 23:55 hs**. Será dada uma tolerância de atraso sem desconto de nota para entregues até as 8:00 da manhã do dia 07/12/09 (2ª-feira).