

```

#if ! defined( DESENHO_ )
#define DESENHO_
/*****
*
*  $MCD Módulo de definição: DES  Módulo Desenho
*
*  Arquivo gerado:           DESENHO.H
*  Letras identificadoras:   DES
*
*  Nome da base de software:  Arcabouço para a automação de testes de
programas redigidos em C
*
*  Projeto: Trabalho 2 - Programação Modular
*  Autores:  GB - Gustavo Bach
*           JG - João Lucas Gardenberg
*           MV - Michelle Valente
*
*  $HA Histórico de evolução:
*       Versão  Autor           Data               Observações
*       1      GB,JG,MV    06/abr/2014 início  desenvolvimento
*
*  $ED Descrição do módulo
*
*
*****/

#if defined( DESENHO_OWN )
#define DESENHO_EXT
#else
#define DESENHO_EXT extern
#endif

/***** Declarações exportadas pelo módulo *****/

/* Variáveis exportadas pelo módulo */

int AlturaMatriz ;
int LarguraMatriz ;
int LinhaMatrizValores ;
int ColunaMatrizValores ;

/* Tipo referência para um desenho */

typedef struct tgDesenho * ptDesenho ;

/*****
*
*  $TC Tipo de dados: DES Condições de retorno
*
*****/

```

```

*
* $ED Descrição do tipo
*     Condições de retorno das funções do valor
*
*****/

typedef enum {

    DES_CondRetOK = 0,
        /* Concluiu corretamente */

    DES_CondRetFaltouMemoria,
        /* Faltou memória ao tentar criar um elemento de valor */

    DES_CondRetFimColunas,
        /* Foi atingido o fim das colunas da matriz */

    DES_CondRetFimLinhas,
        /* Foi atingido o fim das linhas da matriz */

    DES_CondRetDesenhoNaoExiste
        /* O desenho não existe */

} DES_tpCondRet ;

/*****
*
* $FC Função: DES &Gerar Matriz
*
* $ED Descrição da função
*     Gera a matriz do jogo.
*
*
* $EP Parâmetros
*     pDesenho - Ponteiro para o campo desenho a ser criado
*     Altura - altura da matriz a ser criada
*     Largura - largura da matriz a ser criada
*
* $FV Valor retornado
*
*
* Assertivas de entradas esperadas
*     Altura e largura são números inteiros maiores que zero.
*
* Assertivas de saídas esperadas
*
*
*****/

```

```

DES_tpCondRet DES_GerarMatriz( ptDesenho * pDesenho,
                                int Altura,
                                int Largura                ) ;

/*****
*
* $FC Função: DES &Alterar Estado Correto
*
* $ED Descrição da função
*   Altera EstadoCorreto da célula
*
*
* $EP Parâmetros
*   pDesenho - Ponteiro para o campo desenho.
*   Linha - linha da célula;
*   Coluna - Coluna da célula;
*   EstadoCorreto - estado correto da célula
*
* $FV Valor retornado
*
*
* Assertivas de entradas esperadas
*   Linha e Coluna são números inteiros maiores que zero.
*
* Assertivas de saídas esperadas
*
*
*****/

DES_tpCondRet DES_AlterarEstadoCorreto( ptDesenho * pDesenho,
                                         int Linha,
                                         int Coluna                ) ;

/*****
*
* $FC Função: DES &Alterar Estado Atual
*
* $ED Descrição da função
*   Altera EstadoCorreto da célula
*
*
* $EP Parâmetros
*   pDesenho - Ponteiro para o campo desenho.
*   Linha - linha da célula;
*   Coluna - Coluna da célula;
*   EstadoCorreto - estado correto da célula
*
* $FV Valor retornado

```

```

*
*
* Assertivas de entradas esperadas
*     Linha e Coluna são números inteiros maiores que zero.
*
* Assertivas de saidas esperadas
*
*
*****/

DES_tpCondRet DES_AlterarEstadoAtual( ptDesenho * pDesenho,
                                     int Linha,
                                     int Coluna
                                     ) ;

/*****
*
* $FC Função: DES &Gerar Listas
*
* $ED Descrição da função
*     Gera as listas de valores do desenho.
*
*
* $EP Parâmetros
*     pDesenho - Ponteiro para o campo desenho.
*
*
* $FV Valor retornado
*
*
* Assertivas de entradas esperadas
*     pDesenho != NULL
*
* Assertivas de saidas esperadas
*
*
*****/

DES_tpCondRet DES_GerarListas( ptDesenho * pDesenho ) ;

/*****
*
* $FC Função: DES &Mostrar Desenho
*
* $ED Descrição da função
*     Mostra no prompt de comando o desenho.
*
*
* $EP Parâmetros
*     pDesenho - Ponteiro para o campo desenho.
*
*

```

```

*   $FV Valor retornado
*
*
*   Assertivas de entradas esperadas
*       pDesenho != NULL
*
*   Assertivas de saidas esperadas
*
*
*****/

    DES_tpCondRet DES_MostrarDesenho( ptDesenho * pDesenho ) ;

#undef DES_EXT

/***** Fim do módulo de definição: DES  Módulo Desenho *****/

#else
#endif

```