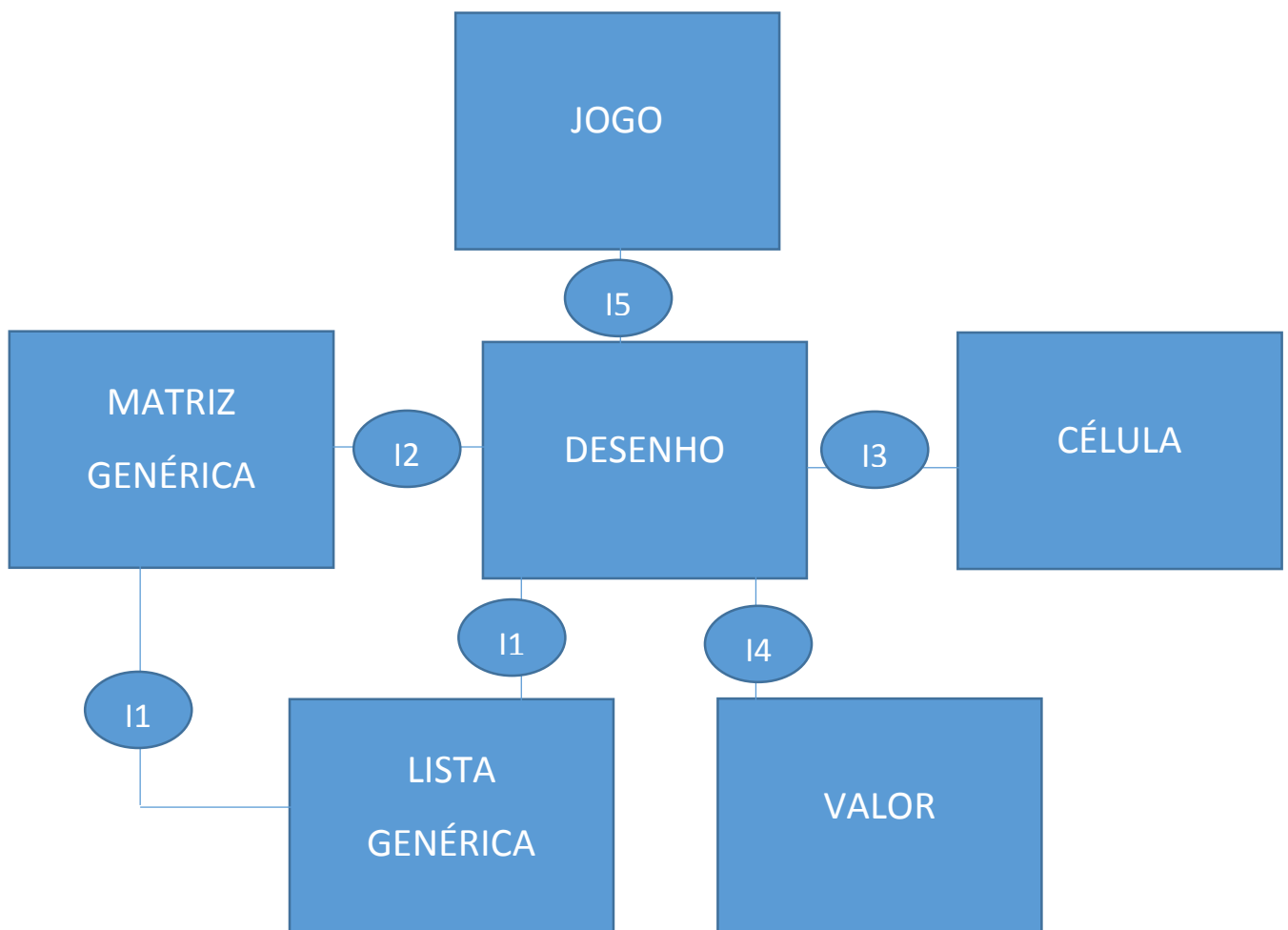


Alunos: Leandro Morgado  
Maria Beatriz Vaz  
Patrick Sava

# PROGRAMAÇÃO MODULAR

## TRABALHO 2

### Arquitetura:



#### Interface I1:

```
TpLista * LST_CriarLista( void );
void LST_EsvaziarLista( TpLista * pLista, void ( * ExcluirValor )(void * pDado));
LST_tpCondRet LST_DestruirLista( TpLista * pLista, void ( * ExcluirValor) ( void
* pDado) );
LST_tpCondRet LST_InserirNovoNoInicio( TpLista * pLista , void * informacao );
LST_tpCondRet LST_InserirNovoNoFim( TpLista * pLista , void * informacao );
LST_tpCondRet LST_InserirNoAntes( TpLista * pLista , void * informacao );
LST_tpCondRet LST_InserirNoApos( TpLista * pLista , void * informacao );
LST_tpCondRet LST_LiberarNoCorrente( TpLista * Lista, void ( * ExcluirValor)
(void * pDado) );
void * LST_ObterValor( TpLista * pLista );
LST_tpCondRet LST_AlterarValor( TpLista * pLista , void * informacao );
LST_tpCondRet LST_Busca( TpLista * pLista , void * informacao , int
(Compara)(void *pDado , void * pDado2) );
LST_tpCondRet LST_IrInicio( TpLista * pLista );
LST_tpCondRet LST_IrFim( TpLista * pLista );
LST_tpCondRet LST_Avancar( TpLista * pLista, int numElem );
```

#### Interface I2:

```
TpMatriz* MAT_CriarMatriz( int linhas, int colunas );
MAT_tpCondRet MAT_InserirNovoElemento(TpMatriz* matriz, void* novo);
void* MAT_ObterElemento( TpMatriz* matriz, int linha, int coluna);
MAT_tpCondRet MAT_AlterarValor( TpMatriz* matriz, void* elem_subst, int linha,
int coluna);
MAT_tpCondRet MAT_DestruirMatriz( TpMatriz* matriz, void ( * ExcluirValor) ( void
MAT_tpCondRet MAT_EsvaziarMatriz( TpMatriz* matriz, void ( * ExcluirValor) ( void
* pDado) );
```

#### Interface I3:

```
Celula* CEL_CriaCelula( int marcacao_esperada, int marcacao_atual );
int CEL_MarcacaoEsperada( Celula* cell);
int CEL_MarcacaoAtual( Celula* cell);
CEL_tpCondRet CEL_AlteraMarcacao( Celula* cell);
CEL_tpCondRet CEL_DestruirCelula( Celula* cell);
```

#### Interface I4:

```
TpValor * VAL_CriarValor ( void );
VAL_tpCondRet VAL_AlterarQntdMarcados ( TpValor * Valor , int quantidade );
VAL_tpCondRet VAL_IncrementarQntdPintados ( TpValor * Valor );
VAL_tpCondRet VAL_DecrementarQntdPintados ( TpValor * Valor );
int VAL_ObterMarcados ( TpValor * Valor );
int VAL_ObterPintados ( TpValor * Valor );
VAL_tpCondRet VAL_ResetarValor ( TpValor * Valor );
VAL_tpCondRet VAL_DestruirValor ( TpValor * Valor );
```

#### Interface I5:

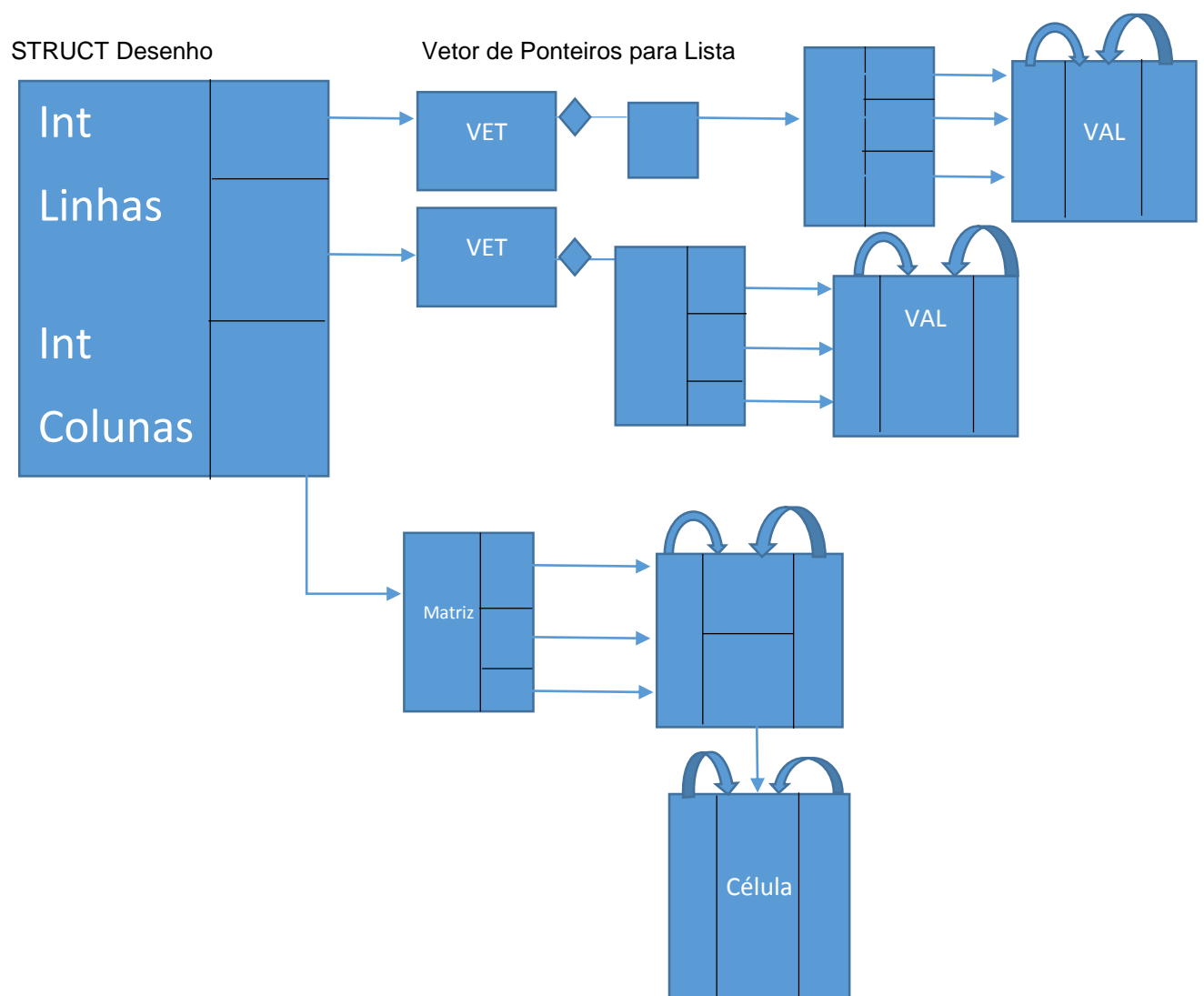
```
DES_tpCondRet DES_IniciaDesenho(unsigned int NumLinhas, unsigned int NumColunas);
DES_tpCondRet DES_IniciaDesenhoVazio( unsigned int NumLinhas, unsigned int
NumColunas );
DES_tpCondRet DES_IniciaDesenhoProjetado( FILE * ArquivoDesenho );
DES_tpCondRet DES_RetomaUltimoJogoSalvo( void );
DES_tpCondRet DES_DestroiDesenho( void );
```

```

DES_tpCondRet DES_AlteraMarcacaoCoordenada( unsigned int Coord_X, unsigned int
Coord_Y );
DES_tpCondRet DES_AtivaDica ( void );
DES_tpCondRet DES_ImprimeMatrizJogo( void );
DES_tpCondRet DES_GravaJogoAtual( void );
DES_tpCondRet DES_GravaMatrizProjetada( char* NomeArquivo );
int DES_DesenhoFinalizado ( void );

```

## Modelo Físico:



## Assertivas Estruturais:

### Lista Duplamente Encadeada:

Contém:

- Cabeça da lista
- Nó da lista

- Seja pLista um ponteiro para a cabeça da Lista e pElem um ponteiro para o nó da lista.
- Se pElem -> pProx != NULL, então pElem -> pProx -> pAnt == pElem
- Se pElem -> pAnt != NULL, então pElem -> pAnt -> pProx == pElem
- Se pLista -> numElementos == 0, então pLista->pNoCorrente == NULL e  
pLista -> pOrigemLista == NULL e  
pLista -> pFimLista == NULL
- Se pLista -> numElementos > 0, então pLista->pNoCorrente != NULL
- Se pLista -> numElementos == 1,  
então pLista ->pNoCorrente == pLista-> pNoOrigem e  
pLista -> pNoCorrente == pLista -> pNoFim

### Matriz:

Contém:

- Número de Linhas
- Número de Colunas
- Ponteira para lista duplamente encadeada

- Seja pMatriz um ponteiro para uma matriz, então pMatriz->numColunas > 0 e  
pMatriz -> numLinhas > 0
- Se pMatriz não é nula, então pMatriz->pLista != NULL
- Para pMatriz -> pLista valem as assertivas da Lista Duplamente Encadeada