

## Assertivas Estruturais:

### LISTA

Seja lista um ponteiro para estrutura cabeça da lista.

Seja no um ponteiro para estrutura nó da lista.

Se `no->pProx != NULL`, então `no->pProx->pAnt == no`

Se `no->pAnt != NULL`, então `no->pAnt->pProx == no`

Se `lista->numElem == 0`, então

- `lista->pElemCorr == NULL`
- `lista->pOrigemLista == NULL`
- `PFimLista == NULL`

Se `lista->numElem > 0`, então `lista->pElemCorr != NULL`

Se `lista->numElem == 1`, então

- `lista->pElemCorr == lista->pOrigemLista`
- `lista->pOrigemLista == lista->pFimLista`

### BARALHO

Seja Bar uma estrutura Baralho.

Seja deck um ponteiro para estrutura para uma lista de cartas.

Seja qtd o número de cartas em um deck.

Se `Bar->deck == NULL`, então `Bar->qtd = 0`

Se `Bar->deck != NULL`, então `Bar->qtd > 0`

Se `Bar->qtd == 40` (com baralho não embaralhado) então `Bar->deck ==` Lista de cartas em ordem:

(PAUS – COPAS – ESPADAS – OUROS // A, 2, 3, 4, 5, 6, 7, J, Q, K)

Ao chamar a função `BAR_Embaralhar(Bar)`, então `Bar->qtd = 40` e `Bar->deck` é uma lista de cartas em ordem aleatória.

Como `Bar->deck` é uma lista, ele segue as assertivas da **LISTA** mencionadas acima.