



# Pontifícia Universidade Católica do Rio de Janeiro

Alunos

1711799 - Fernando Tancini  
1710635 - Júlia Rocha  
1811208 - Suemy Inagaki

Professor

Flávio Heleno Bevilacqua e Silva

Rio de Janeiro, 17 de Junho de 2019

## Lista duplamente encadeada genérica com cabeça (Original)

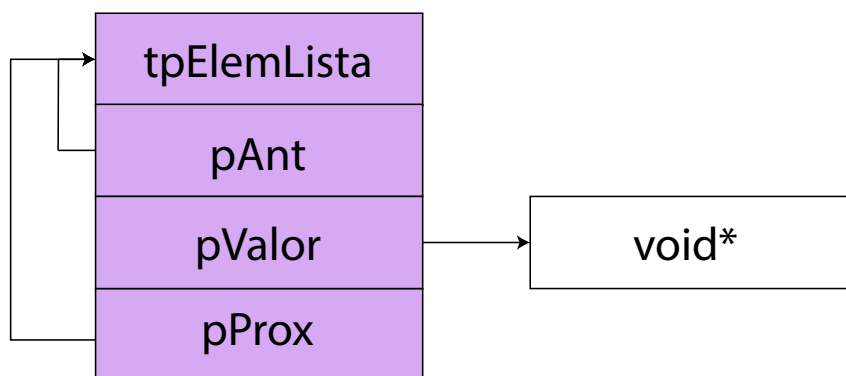
### Estrutura: tpElemLista

- pAnt -> Ponteiro para o elemento anterior da lista
- pProx -> Ponteiro para o próximo elemento da lista
- pValor-> Conteúdo do elemento

### Assertivas Estruturais:

- Se pCorr -> pProx != NULL então pCorr -> pProx -> -> pAnt = pCorr ou se pCorr->pAnt != NULL então pCorr->pAnt->pProx = pCorr.
- pProx só será NULL caso o elemento seja o último ou o único da lista.
- pAnt só será NULL caso o elemento seja o primeiro ou o único da lista.

### Modelo:



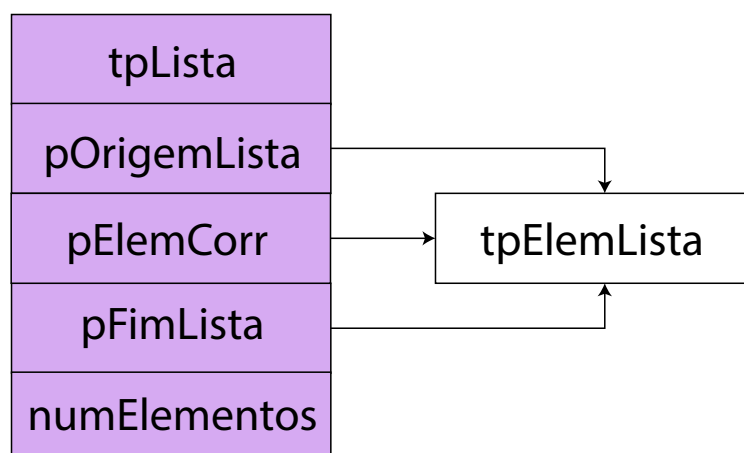
## Estrutura: tpLista

- pOrigemLista -> Ponteiro para o primeiro elemento da lista.
- pElemCorr -> Ponteiro para o elemento corrente da lista.
- pFimLista->Ponteiro para o último elemento da lista.
- NumElementos->Indica quantos elementos estão presentes na lista.

## Assertivas Estruturais:

- pOrigem Lista só será NULL caso a lista esteja vazia, assim NumElementos = 0
- pFimLista só será NULL caso a lista esteja vazia, , assim NumElementos = 0.
- Se pFimLista == pOrigemLista, logo só existe um único elemento na lista, assim NumElementos = 1.
- O elemento anterior ao primeiro elemento é inexistente, ou seja, pOrigemLista-> pAnt == NULL.
- O próximo elemento do último elemento é inexistente, ou seja, pFimLista-> pProx == NULL.
- pElemCorr deve apontar para um elemento da lista ou deve ser NULL.

## Modelo:



## Lista Genérica Atualizada (auto-verificável)

### Estrutura: tpElemLista

- pAnt -> Ponteiro para o elemento anterior da lista.
- pProx -> Ponteiro para o próximo elemento da lista.
- pValor-> Conteúdo do elemento.
- pCabeca-> Ponteiro para o elemento cabeça da lista.
- tipo-> o tipo do elemento.

### Assertivas Estruturais:

- Se pCorr -> pProx != NULL então pCorr -> pProx -> pAnt = pCorr ou se pCorr->pAnt != NULL então pCorr->pAnt->pProx = pCorr.
- pProx só será NULL caso o elemento seja o último ou o único da lista.
- pAnt só será NULL caso o elemento seja o primeiro ou o único da lista.
- Ponteiro pCabeca deve apontar para uma estrutura Lista, que deverá ser a mesma apontada pelos modelos relacionados, caso existam (pProx e pAnt).
- Tipo deve ser igual ao Tipo da estrutura tpLista apontada pelo ponteiro pCabeca.

### Estrutura: tpLista

- pOrigemLista -> Ponteiro para o primeiro elemento da lista.
- pElemCorr -> Ponteiro para o elemento corrente da lista.
- pFimLista->Ponteiro para o último elemento da lista.
- NumElementos->Indica quantos elementos estão presentes na lista.
- tipo-> o tipo do elemento.

### Assertivas Estruturais:

- pOrigem Lista só será NULL caso a lista esteja vazia, assim NumElementos = 0
- pFimLista só será NULL caso a lista esteja vazia, , assim NumElementos = 0.
- Se pFimLista == pOrigemLista, logo só existe um único elemento na lista, assim NumElementos = 1.

- O elemento anterior ao primeiro elemento é inexistente, ou seja, `pOrigemLista-> pAnt == NULL`.
- O próximo elemento do último elemento é inexistente, ou seja, `pFimLista-> pProx == NULL`.
- `pElemCorr` deve apontar para um elemento da lista ou deve ser `NULL`.
- **Tipo deve ser igual ao Tipo da estrutura `tpElemLista` apontado na Lista.**

### Modelo:

