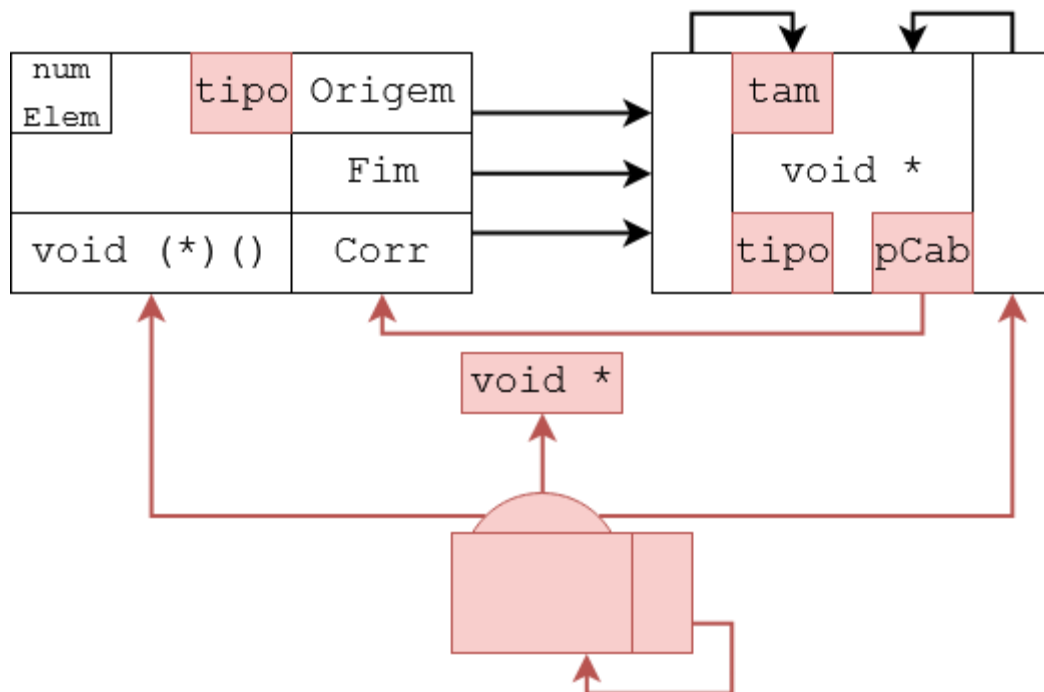


## Modelo: Lista Duplamente Encadeada Genérica com Cabeça



Os campos e ponteiros em preto fazem parte da estrutura da lista.

Os campos e ponteiros em **vermelho** foram incluídos para transformar a lista em uma estrutura auto verificável. Aparecem somente na compilação com `_DEBUG` definido.

### Assertivas Estruturais

- Se `pCorr->pAnt != NULL` então `pCorr->pAnt->pProx == pCorr`
- Se `pCorr->pProx != NULL` então `pCorr->pProx->pAnt == pCorr`
- Os campos `tipo`, tanto da cabeça quanto do nó apontam para strings que indicam o tipo apontado pelo nó. Assim, os dois campos devem ser iguais. Todos os elementos da lista devem apontar para o mesmo tipo.
- O campo `tam` do nó é um inteiro que indica o tamanho do conteúdo apontado pelo nó, em bytes.
- O campo `pCab` do nó aponta para a cabeça da lista a qual aquele nó pertence.
- `pCorr->pAnt == NULL` se e somente se o nó é o primeiro da lista.
- `pCorr->pProx == NULL` se e somente se o nó é o último da lista.
- `pCorr == NULL` se e somente se a lista está vazia (logo `pOrigem` e `pFim` também serão `NULL`)
- Cada elemento da lista de espaços alocados (em vermelho) aponta para a cabeça da lista, um elemento ou o conteúdo apontado pelo elemento caso esse tenha sido criado com `malloc`.
- Na lista de espaços alocados cada elemento aponta para o próximo.