Especificação do Tipo Abstrato de Dados (TAD) - Lista de Nomes

**Especificação do TAD Lista de Nomes**

Este documento apresenta a especificação do Tipo Abstrato de Dados (TAD) para uma lista de nomes, contendo as operações solicitadas. Cada operação é descrita em ter- mos de entrada, pré-condição, processo, saída e pós-condição, sem foco em detalhes de implementação.

# Operação: insere\_elemento

## Entrada:

* + Uma lista l de nomes.
  + Uma string nome representando o nome a ser inserido.

## Pré-condição:

* + A lista l deve ser válida (não nula).
  + A lista l não deve estar cheia.
  + A string nome deve ser válida (não nula).

## Processo:

* + Insere o nome nome no final da lista l.

## Saída:

* + Um inteiro: 1 se a inserção for bem-sucedida, 0 caso contrário (lista nula ou cheia).

## Pós-condição:

* + O nome nome é adicionado ao final da lista l, se a inserção for bem-sucedida.
  + O tamanho da lista é incrementado em 1, se a inserção for bem-sucedida.
  + Se a inserção falhar, a lista permanece inalterada.

# Operação: remove\_elemento

## Entrada:

* + Uma lista l de nomes.
  + Uma string nome representando o nome a ser removido.

## Pré-condição:

* + A lista l deve ser válida (não nula).
  + A lista l não deve estar vazia.
  + A string nome deve ser válida (não nula).

## Processo:

* + Busca o nome nome na lista l.
  + Se encontrado, remove o nome e reorganiza a lista para preencher o espaço vazio.

## Saída:

* + Um inteiro: 1 se a remoção for bem-sucedida, 0 caso contrário (lista nula, vazia ou nome não encontrado).

## Pós-condição:

* + Se a remoção for bem-sucedida, o nome nome não estará mais na lista.
  + O tamanho da lista é decrementado em 1, se a remoção for bem-sucedida.
  + Se a remoção falhar, a lista permanece inalterada.

# Operação: imprimir\_lista

## Entrada:

* + Uma lista l de nomes.
  + Um array de strings lista com capacidade para armazenar os nomes.

## Pré-condição:

* + A lista l deve ser válida (não nula).
  + A lista l não deve estar vazia.
  + O array lista deve ser válido (não nulo) e ter espaço suficiente para armazenar todos os nomes da lista.

## Processo:

* + Copia todos os nomes da lista l para o array lista.

## Saída:

* + Um inteiro representando o número de nomes copiados (tamanho da lista), ou

0 se a operação falhar (lista nula ou vazia).

* + O array lista contém os nomes da lista l, se a operação for bem-sucedida.

## Pós-condição:

* + O array lista contém os nomes da lista l, na mesma ordem, se a operação for bem-sucedida.
  + A lista l permanece inalterada.
  + Se a operação falhar, o array lista permanece inalterado.

# Operação: pegar\_elemento

## Entrada:

* + Uma lista l de nomes.
  + Uma string lista onde o nome será armazenado.
  + Um inteiro elemento representando a posição do nome desejado (base 0).

## Pré-condição:

* + A lista l deve ser válida (não nula).
  + A lista l não deve estar vazia.
  + A posição elemento deve ser válida (0 *≤* elemento *<* tamanho da lista).
  + A string lista deve ser válida (não nula) e ter espaço suficiente para ar- mazenar o nome.

## Processo:

* + Recupera o nome na posição elemento da lista l e o copia para a string lista.

## Saída:

* + Um inteiro: 1 se a recuperação for bem-sucedida, 0 caso contrário (lista nula, vazia ou posição inválida).
  + A string lista contém o nome na posição elemento, se a operação for bem- sucedida.

## Pós-condição:

* + A string lista contém o nome da posição elemento, se a operação for bem- sucedida.
  + A lista l permanece inalterada.
  + Se a operação falhar, a string lista permanece inalterada.

# Operação: apagar

## Entrada:

* + Um ponteiro l para uma lista de nomes.

## Pré-condição:

* + Nenhuma (a operação lida com casos de ponteiro nulo).
* **Processo:**

# Operação: esvaziar\_lista

## Entrada:

∗ Uma lista l de nomes.

## Pré-condição:

∗ A lista l deve ser válida (não nula).

## Processo:

∗ Remove todos os elementos da lista l, deixando-a vazia.

## Saída:

∗ Nenhuma.

## Pós-condição:

∗ A lista l está vazia (tamanho igual a 0).

∗ Se a operação falhar (lista nula), a lista permanece inalterada.