

# Fila Encadeada

Davi Teixeira - 118309

Marcelo Danigno - 116509

Michel Neves - 116510

# Fila Encadeada

Na Fila Encadeada os nodos **não** estão alocados em sequência, isto é o próximo nodo não ocupa o endereço seguinte do atual, mas sim um endereço "aleatório" na memória.

# Fila - Características

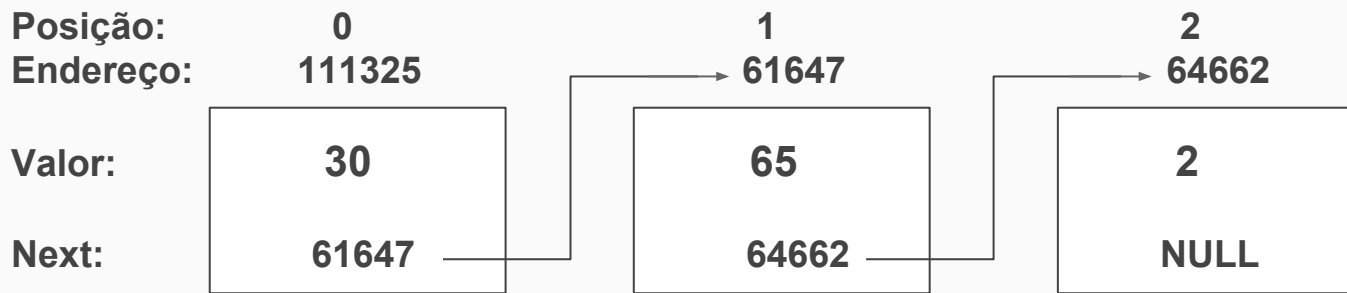
Em uma fila o primeiro valor que entra também será o primeiro a sair (FIFO - First In First Out).

## **É Proibido:**

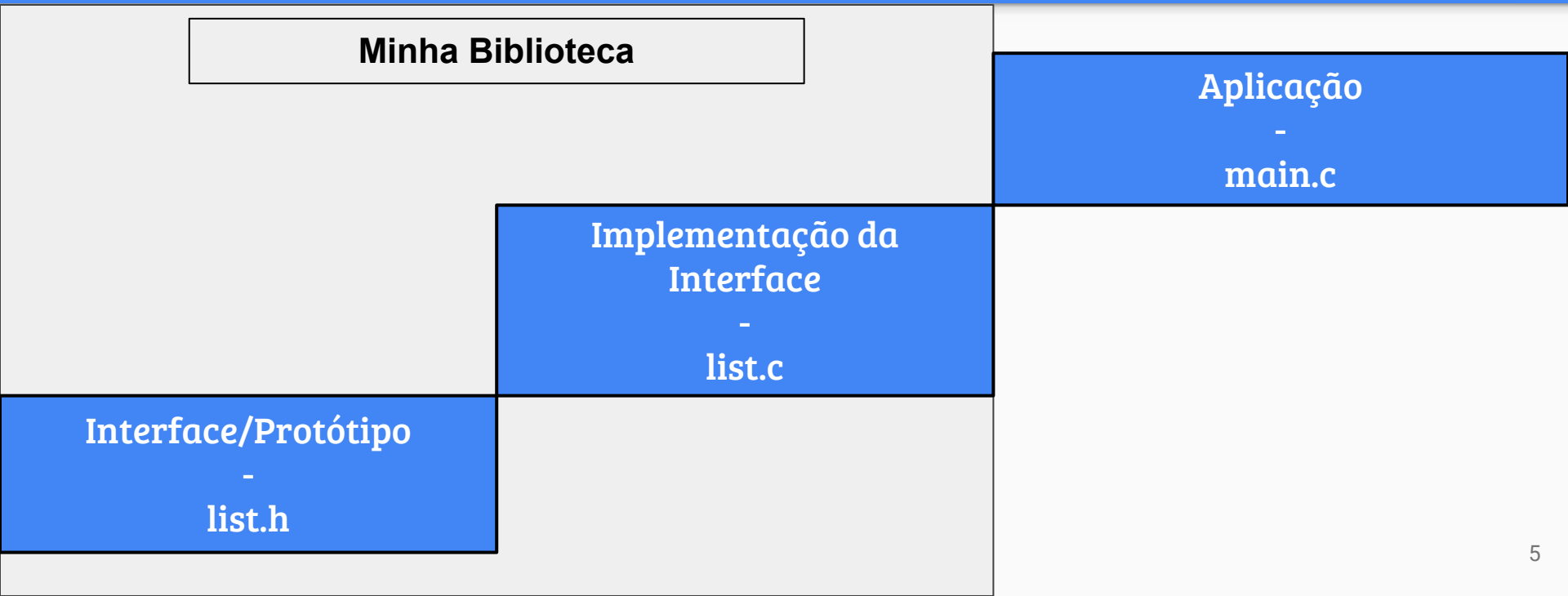
- Inserir novo Nodo em qualquer lugar que não seja no fim.
- Consultar Nodo de qualquer lugar que não seja o do começo.
- Deletar Nodo de qualquer lugar que não seja o do começo.

# Fila Encadeada na Teoria

A fila encadeada funciona de modo que o próximo nodo tem seu endereço indicado em um campo do nodo atual.



# Descrição do TAD



# Interface

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <math.h>
```

```
/* Tipo Node abstrai a estrutura do nodo */
```

```
typedef struct node Node;
```

```
/* Aloca e retorna um nodo vazio */
```

```
Node* Initialize();
```

```
/* Adiciona um nodo no final da fila */
```

```
Node* Add(Node *list, int value);
```

```
/* Retira o primeiro nodo da fila */
```

```
void Pop(Node **list);
```

```
/* Retira o primeiro nodo da fila e retorna true se conseguiu e false caso contrário */
```

```
bool PopBool(Node **list);
```

```
/* Retira o primeiro nodo da fila e o retorna o seu valor */
```

```
int Top(Node **list);
```

```
/* Mostra o primeiro nodo da fila na tela */
```

```
void Print(Node *list);
```

```
/* Desaloca a memória e destrói a fila */
```

```
Node* Destroy(Node *list);
```

# Aplicação