

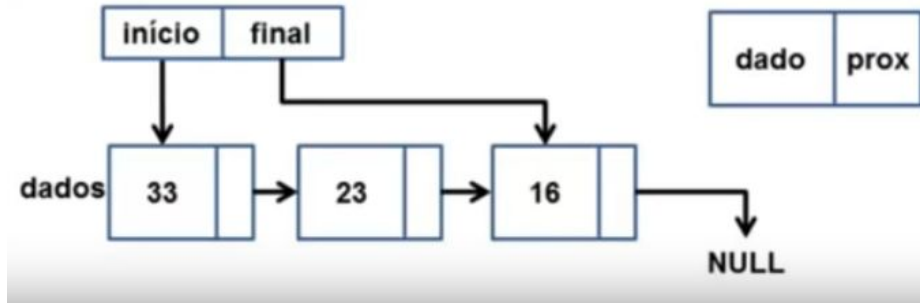
# Filas Dinâmicas

Prof Marlus Dias Silva

# Definição

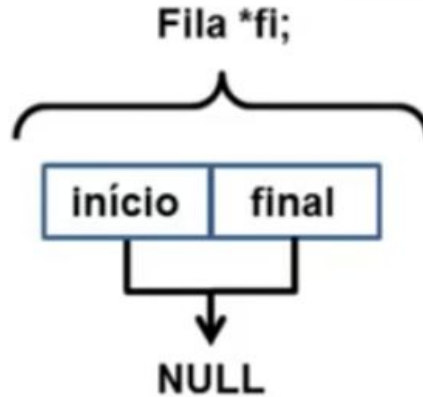
Fila Dinamica -> tipo de "fila" onde cada elemento aponta para o seu sucessor na "fila"

OBS: Usa um "nó descritor" para representar o início e o final da "fila" e uma indicação de final de fila.

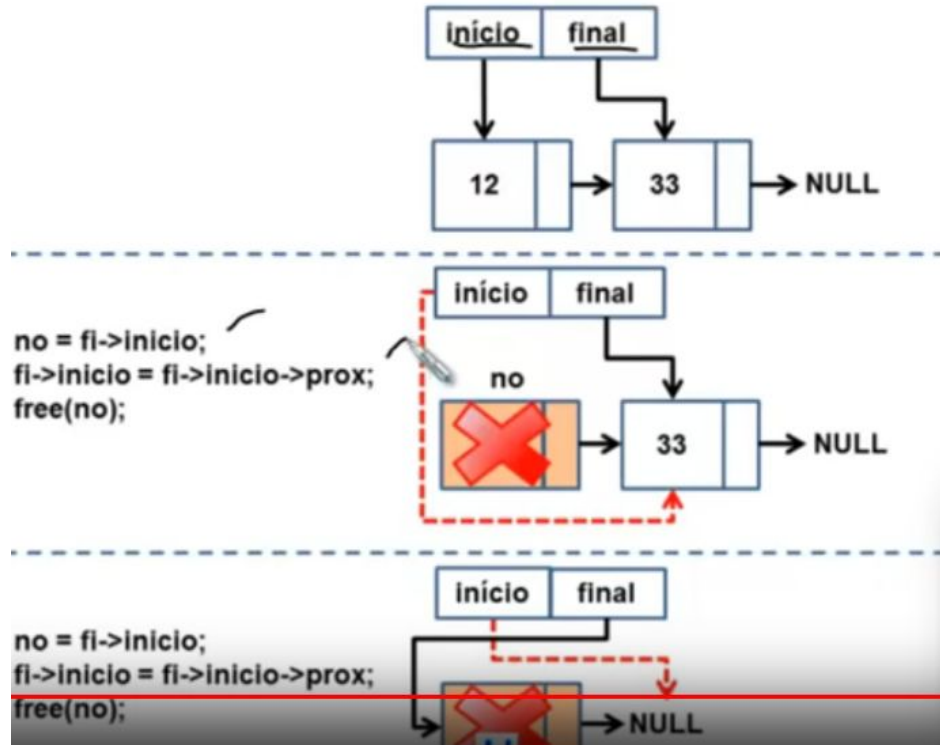


# Criando uma Fila

```
Fila* criar_fila(){  
    Fila *fi = (Fila*) malloc(sizeof(Fila));  
    if(fi != NULL) {  
        fi->final = NULL;  
        fi->inicio = NULL;  
    }  
}
```

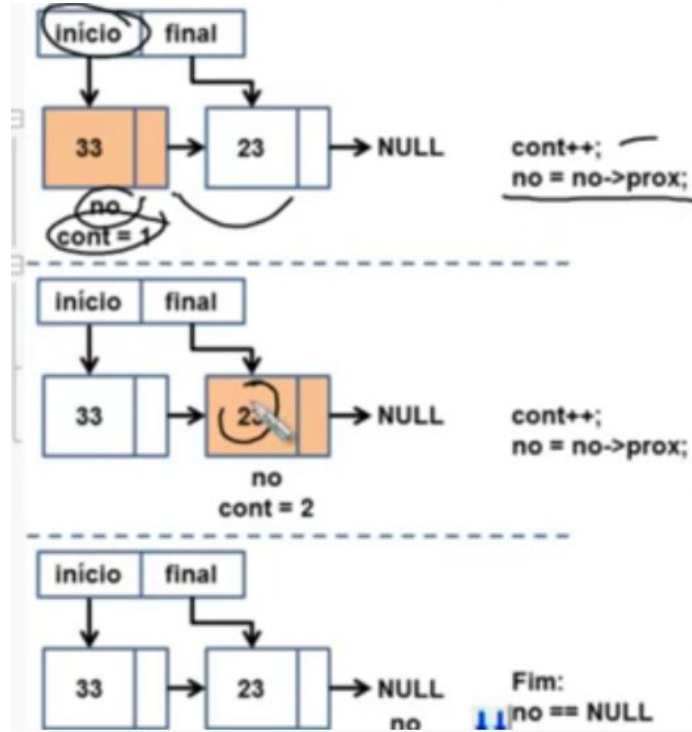


# Librando elementos da Fila



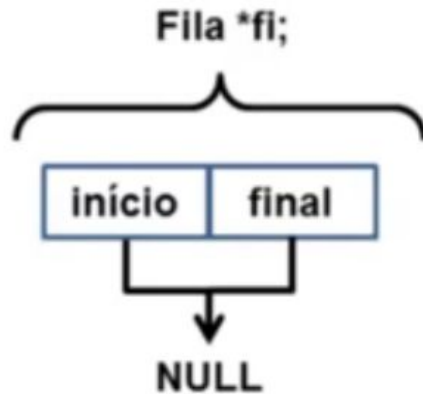
# Verificando o tamanho da Fila

```
int tamanho_fila(Fila *fi){  
    if(fi == NULL) return false;  
    int cont = 0;  
    Elem* no = fi->inicio;  
    while(no != NULL){  
        cont++;  
        no = no->prox;  
    }  
    return cont;  
}
```



# Fila Vazia

```
int fila_vazia(Fila *fl){  
    if(fl == NULL) return false;  
    if(fl->inicio == NULL)  
        return true;  
    return false;  
}
```



# Sentido da Fila

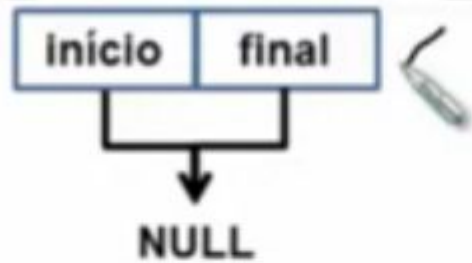
Em uma fila, a inserção sempre acontece no seu final.

Em uma fila, a remoção sempre acontece no seu início.

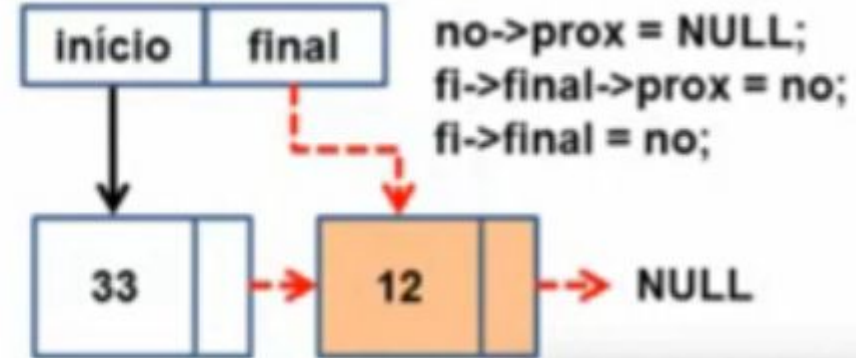
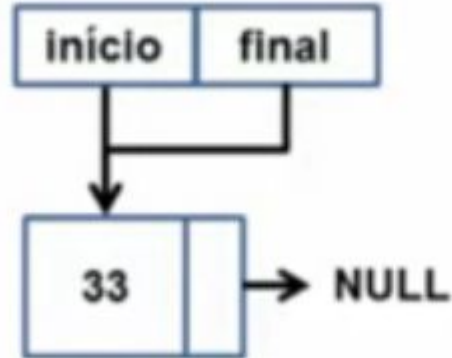


# Inserir elementos na Fila

Inserção em fila vazia



Inserção em fila NÃO vazia





# Inserção de Elementos na Fila

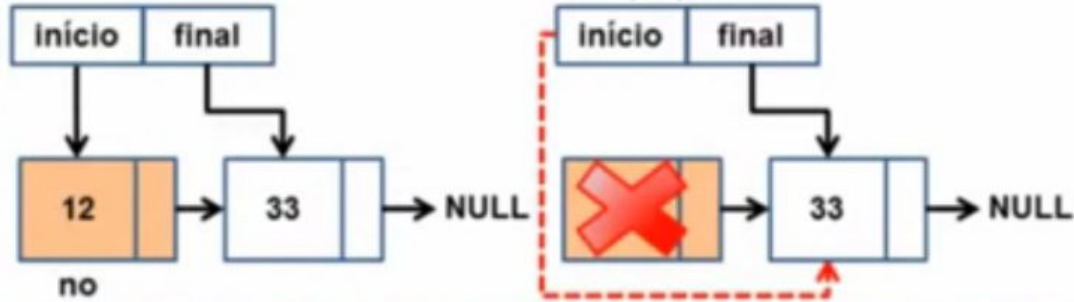
```
int insere_fila(Fila* fi, struct aluno al){  
    if(fi == NULL) return false;  
    Elem *no = (Elem*) malloc(sizeof(Elem));  
    if(no == NULL) return false;  
    no->dados = al;  
    no->prox = NULL;  
    if(fi->final == NULL) // fila vazia  
        fi->inicio = no;  
    else  
        fi->final->prox = no;  
    fi->final = no;  
    return true;  
}
```



# Remoção de elementos na fila

```
int x = remove Fila(fi);
```

fila NÃO  
fica vazia



fila fica  
vazia



# Remove Elementos da fila

```
int remove_fila(Fila* fi){  
    if(fi == NULL) return false;  
    if(fi->inicio == NULL) return false;  
    Elem *no = fi->inicio;  
    fi->inicio = fi->inicio->prox;  
    if(fi->inicio == NULL)  
        fi->final = NULL;  
    free(no);  
    return true;  
}
```

# Consultar elemento na Fila

```
int consulta_fila(Fila *fi, struct aluno *al){  
    if(fi == NULL) return false;  
    if(fi->inicio == NULL) //fila vazia  
        return false;  
    *al = fi->inicio->dados;  
    return true;  
}
```

