

## SCE 182 – Algoritmos e Estruturas de Dados I



# Filas

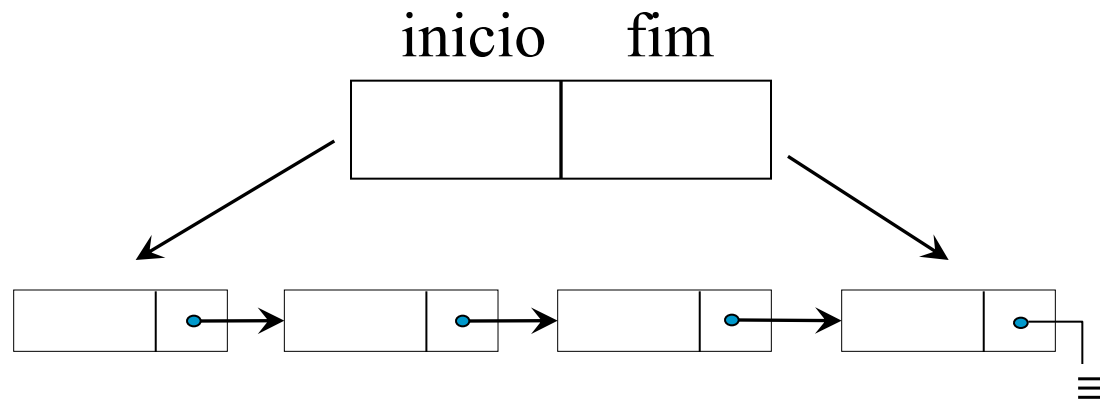
# Implementação Dinâmica

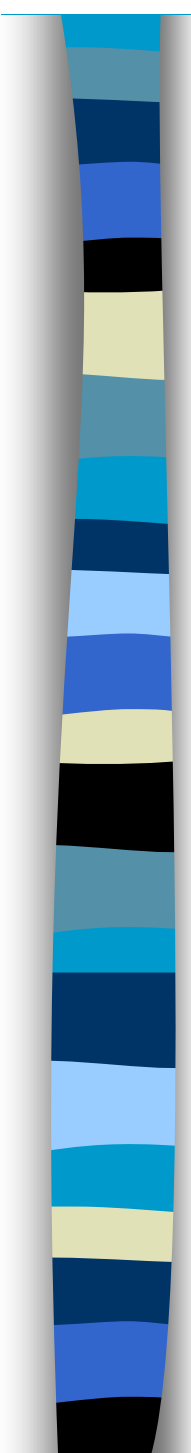
# Implementações de Filas: Dinâmica

```
#define tipo_info int
```

```
typedef struct elem{  
    tipo_info info;  
    struct elem *lig;  
}tipo_elem;
```

```
typedef struct{  
    tipo_elem *inicio;  
    tipo_elem *fim;  
}fila;  
fila q;
```





```
void definir(fila *q){
    /*Cria uma fila vazia. Deve ser usado antes de qualquer
    outra operação*/
    q->inicio = NULL;
    q->fim = NULL;
}

boolean vazia (fila *q){
    /*Retorna true se fila não contém elementos, false caso
    contrário*/
    return (q->inicio == NULL);
}

void tornar_vazia (fila *q){
    /*Reinicializa uma fila existente q como uma fila vazia
    removendo todos os seus elementos.*/
    tipo_elem *ndel, *nextno;

    if(!vazia(q)){
        nextno = q->inicio;
        while (nextno != NULL){
            ndel = nextno;
            nextno = nextno->lig;
            free(ndel);
        }
    }
    definir(q);
}
```

```

boolean inserir (fila *q, tipo_info info){
    /*Adiciona um item no fim da fila q. Retorna true se
    operação realizada com sucesso, false caso contrário*/
    tipo_elem *p;

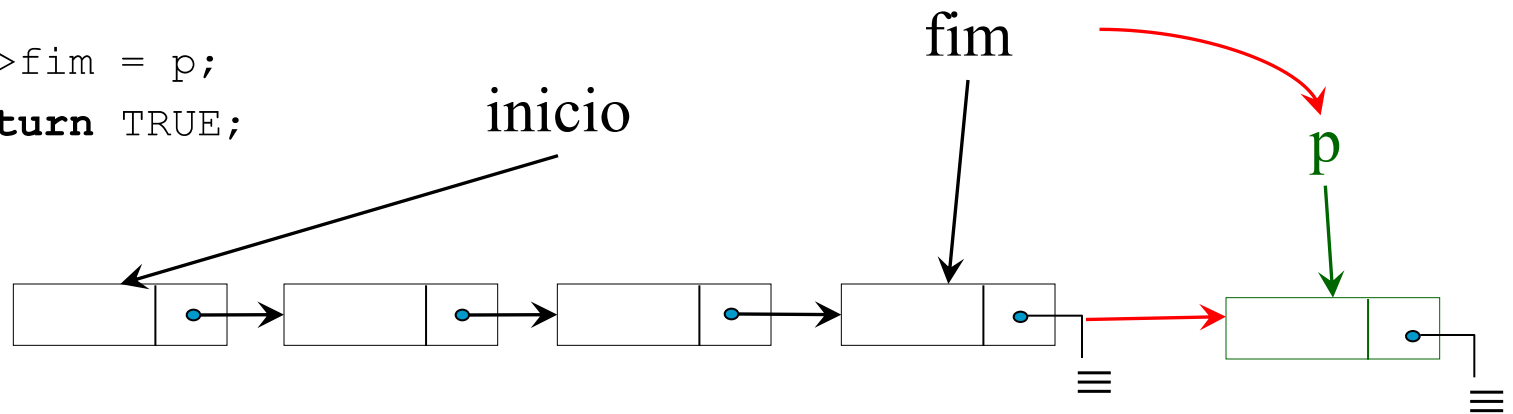
    p = malloc(sizeof(tipo_elem));
    if (p == NULL)
        return FALSE;

    p->info = info;
    p->lig = NULL;

    if (vazia(q))
        q->inicio = p;
    else
        q->fim->lig = p;

    q->fim = p;
    return TRUE;
}

```



```

boolean remover(fila *q, tipo_info *info){
    /*Remove um item do início da fila q. Retorna true se
    operação realizada com sucesso, false caso contrário*/
    tipo_elem *p;

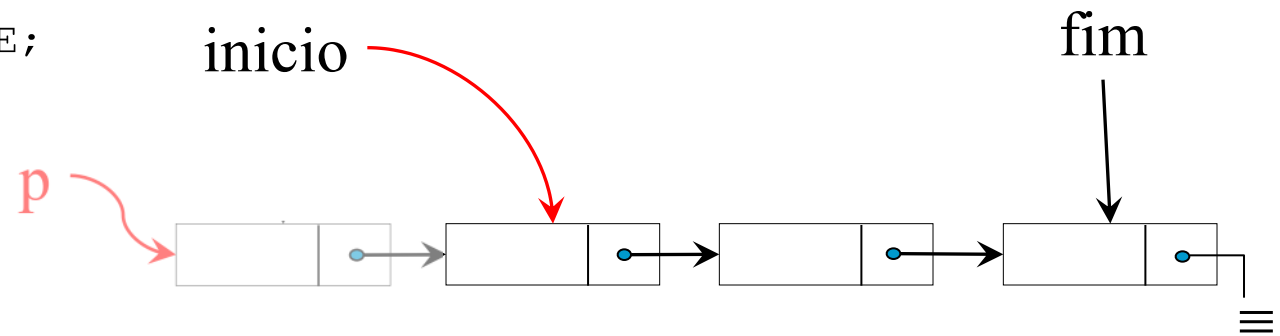
    if (vazia(q))
        return FALSE;

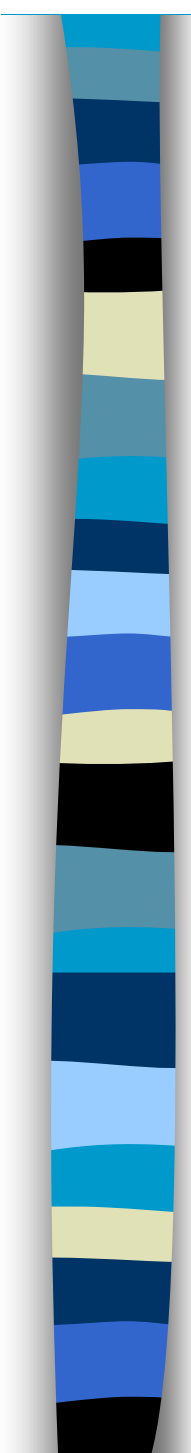
    p = q->inicio;
    *info = p->info;
    q->inicio = p->lig;

    if (q->inicio == NULL)
        q->fim = NULL;

    free(p);
    return TRUE;
}

```





```
int tamanho (fila *q){
    /*Retorna o tamanho da fila*/
    tipo_elem *p;
    int cont = 0;

    p = q->inicio;
    while (p != NULL){
        cont ++;
        p = p->lig;
    }

    return cont;
}
```

```
boolean começo_fila (fila *q, tipo_info *item){
    /*Mostra o começo da fila sem remover o item. Retorna true
    se operação realizada com sucesso, false caso contrário*/
    if (vazia(q))
        return FALSE;

    *item = q->inicio->info;
    return TRUE;
}
```



# Análise do tipo de Representação

- Vantagens da Fila Dinâmica:
  - ocupa espaço estritamente necessário
- Desvantagens da Fila Dinâmica:
  - custos usuais da alocação dinâmica (tempo de alocação, campos de ligação)



## Quando usar

- Representação Estática (Anel):
  - quando fila tiver tamanho pequeno ou seu comportamento for previsível
- Representação Dinâmica:
  - nos demais casos



# Atividade

---

Faça um programa que, utilizando as funções criadas nos exemplos dessa aula crie uma fila F e exiba o seguinte menu de opções:

## **EDITOR DE FILA**

- 1 - INSERIR
- 2 - REMOVER
- 3 - EXIBIR ELEMENTO DO INICIO
- 4 - EXIBIR A FILA
- 5 - ESVAZIAR A FILA
- 6 - SAIR

DIGITE SUA OPÇÃO:

# Bibliografia Básica

---

- ❑ SILVA, Osmar Quirino da. **Estruturas de Dados e Algoritmos Usando C – Fundamentos e Aplicações.** Rio de Janeiro, Ed: Ciência Moderna, 2007.
- ❑ MANZANO, Wilson Y. Yamaturni-São Paulo-SP. **Lógica estruturada para programação de computadores,** Ed. Érica 1997 e 2001.
- ❑ MORAES, Celso Roberto. **Estruturas de Dados e Algoritmos.** Ed. Érica, São Paulo.
- ❑ LOPES, Anita. **Introdução à programação.** Rio de Janeiro: Campus, 2002.

# Bibliografia Complementar

---

- ❑ BENEDUZZI, Humberto M. e METZ, João A. **Lógica e Linguagem de Programação – Introdução ao Desenvolvimento de Software** (1ª edição). Editora do Livro Técnico, 2010
- ❑ FORBELLONE, A. L. V. e Eberspacher, H. F. **Lógica de Programação - a Construção de Algoritmos e Estruturas de Dados** (3ª edição). Pearson, 2005
- ❑ CORMEN, Thomas H. et. al. **Algoritmos: Teoria e Prática**. Editora Campus, 2002.
- ❑ ZIVIANI, Nivio. **Projeto de Algoritmos**. Editora Nova Fronteira, 2004.
- ❑ SEBESTA, Robert W. **Conceitos de Linguagens de Programação**. Bookman, 2001.