

Pilhas e Filas

Estruturas de Dados - Pilhas & Filas

Listas lineares especiais



Disciplina restrita de acesso

Acesso

- consulta
- inserção
- remoção

Disciplina restrita

- acesso permitido somente em alguns nós

Estruturas de Dados - Pilhas & Filas

Listas lineares especiais mais usuais

LIFO Last In First Out

o último componente inserido é o primeiro a ser retirado

Pilha

FIFO First In First Out

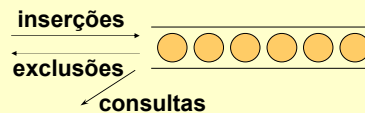
o primeiro componente inserido é também o primeiro a ser retirado

Fila

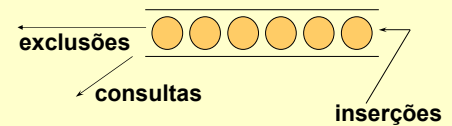
Estruturas de Dados - Pilhas & Filas

Pilhas e Filas

Pilhas:



Filas:

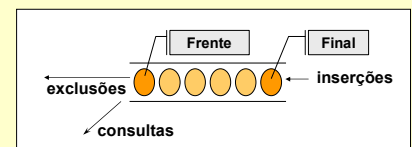


Estruturas de Dados - Pilhas & Filas

Filas Encadeamento

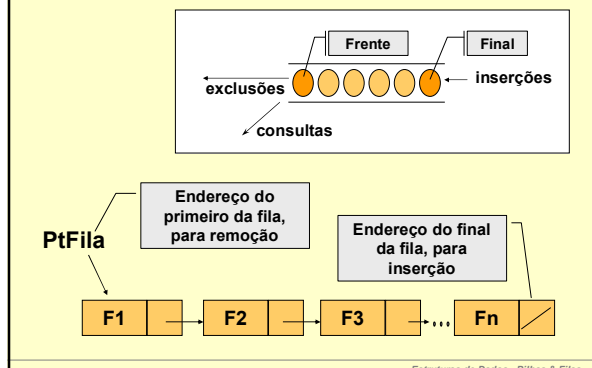
Estruturas de Dados - Pilhas & Filas

Filas encadeadas

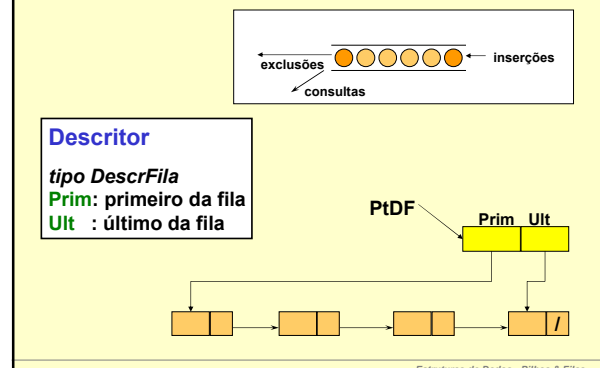


Estruturas de Dados - Pilhas & Filas

Filas encadeadas



Filas encadeadas - com descritor



Especificação da Estrutura de Dados

```
typedef int TipoInfo;
typedef struct s_TipoDFila TipoDFila;
```

```
struct TPtFila{
    TipoInfo dado;
    struct TPtFila *elo;
};

typedef struct TPtFila TipoFila;

struct s_TipoDFila{
    struct TPtFila *prim;
    struct TPtFila *ult;
};
```

TAD

```
TipoDFila* InicializaFila (TipoDFila *PtDFila);
TipoInfo ConsultaFila (TipoDFila *PtDFila);
int InserirFila(TipoDFila **PtDFila, TipoInfo Dado);
int RemoverFila(TipoDFila **PtDFila, TipoInfo *Dado);
void DestroiFila (TipoDFila **PtDFila);
```

Criação de fila encadeada com descritor



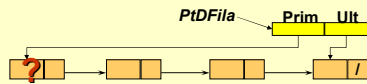
```
TipoDFila* InicializaFila (TipoDFila *PtDFila)
{
    PtDFila = (TipoDFila*) malloc(sizeof(TipoDFila)); //aloca descritor de fila

    if (PtDFila != NULL){ //testa se conseguiu alocar o descritor
        PtDFila->prim = NULL; //inicializa ponteiros
        PtDFila->ult = NULL;
    }
    return PtDFila;
}
```

TAD

```
TipoDFila* InicializaFila (TipoDFila *PtDFila);
TipoInfo ConsultaFila (TipoDFila *PtDFila);
int InserirFila(TipoDFila **PtDFila, TipoInfo Dado);
int RemoverFila(TipoDFila **PtDFila, TipoInfo *Dado);
void DestroiFila (TipoDFila **PtDFila);
```

Consulta à fila encadeada com descritor



```

TipoInfo ConsultaFila (TipoDFila PtDFila)
{
    if (PtDFila->prim == NULL)
        return 0;
    else
        return PtDFila->prim->dado;
}
    
```

Estruturas de Dados - Pilhas & Filas

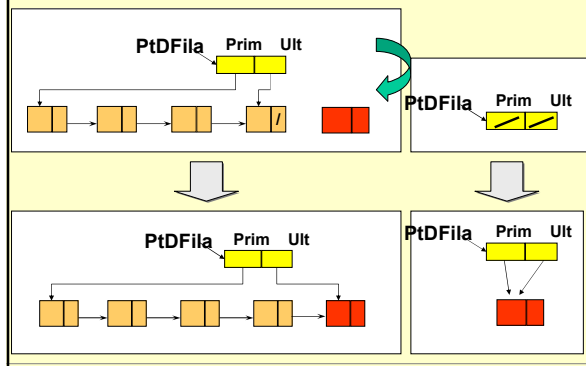
TAD

```

TipoDFila* InicializaFila (TipoDFila *PtDFila);
TipoInfo ConsultaFila (TipoDFila *PtDFila);
int InserirFila(TipoDFila **PtDFila, TipoInfo Dado);
int RemoverFila(TipoDFila **PtDFila, TipoInfo *Dado);
void DestroiFila (TipoDFila **PtDFila);
    
```

Estruturas de Dados - Pilhas & Filas

Inserção em fila encadeada com descritor



Estruturas de Dados - Pilhas & Filas

Inserção em fila encadeada sem descritor

```

int InserirFila(TipoDFila **PtDFila, TipoInfo Dado)
{
    TipoFila *novo;
    if (*PtDFila) { //testa se o descritor foi alocado
        novo = (TipoFila*) malloc (sizeof(TipoFila));
        novo->dado = Dado;
        novo->elo = NULL;
        if ((*PtDFila)->prim == NULL) // vai ser o primeiro da fila
            (*PtDFila)->prim = novo;
        else
            (*PtDFila)->ult->elo = novo; //insere no final
        (*PtDFila)->ult = novo;
        return 1;
    }
    else
        return 0;
}
    
```

Estruturas de Dados - Pilhas & Filas

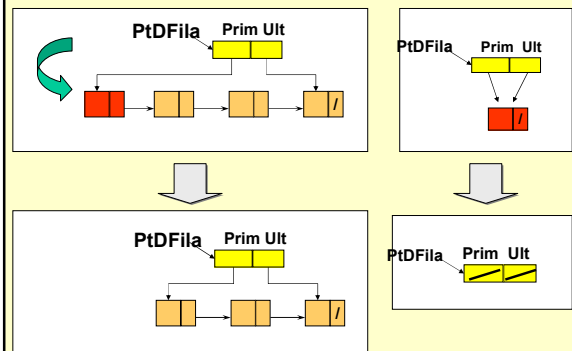
TAD

```

TipoDFila* InicializaFila (TipoDFila *PtDFila);
TipoInfo ConsultaFila (TipoDFila *PtDFila);
int InserirFila(TipoDFila **PtDFila, TipoInfo Dado);
int RemoverFila(TipoDFila **PtDFila, TipoInfo *Dado);
void DestroiFila (TipoDFila **PtDFila);
    
```

Estruturas de Dados - Pilhas & Filas

Remoção de fila encadeada com descritor



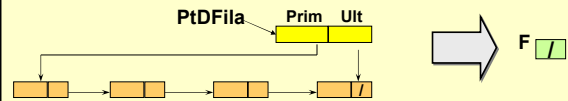
Estruturas de Dados - Pilhas & Filas

Remoção de fila encadeada com descritor

```
int RemoverFila(TipoDFila **PtDFila, TipoInfo *Dado)
{
    TipoFila *ptaux;
    if (*PtDFila) { //testa se o descritor foi alocado
        if ((*PtDFila)->prim != NULL) { // testa se tem algum elemento na fila
            ptaux = (*PtDFila)->prim;
            *Dado = (*PtDFila)->prim->dado;
            (*PtDFila)->prim = (*PtDFila)->prim->elo;
            free(ptaux);
            if ((*PtDFila)->prim == NULL) //testa se a fila ficou vazia
                (*PtDFila)->ult = NULL;
            return 1;
        }
        else
            return 0;
    }
}
```

Estruturas de Dados - Pilhas & Filas

Destruição de fila encadeada com descritor



```
void DestroiFila (TipoDFila **PtDFila)
{
    TipoFila *ptaux;
    if (*PtDFila) {
        while ((*PtDFila)->prim != NULL)
        {
            ptaux = (*PtDFila)->prim; //guarda o endereço do primeiro
            (*PtDFila)->prim = (*PtDFila)->prim->elo; //o próximo passa a ser o topo
            free(ptaux); //libera o que estava no topo
        }
        free(*PtDFila);
        *PtDFila = NULL;
    }
}
```

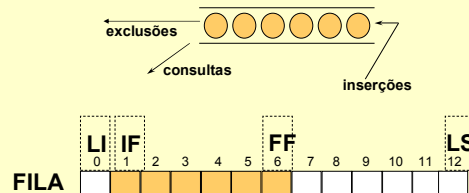
Estruturas de Dados - Pilhas & Filas

Filas

Contigüidade física

Estruturas de Dados - Pilhas & Filas

Fila implementada sobre arranjo

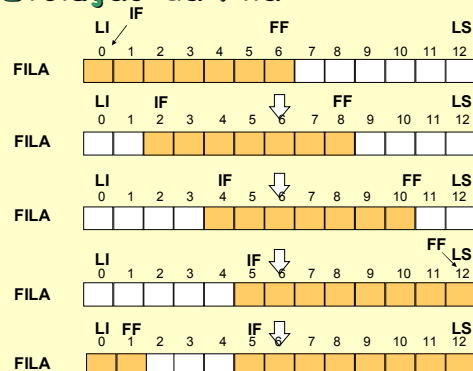


LI : limite inferior da área
IF : início da fila
FF : final da fila
LS : limite superior da área

Fila vazia
IF = -1

Estruturas de Dados - Pilhas & Filas

Evolução da Fila



Estruturas de Dados - Pilhas & Filas

Operações sobre Filas

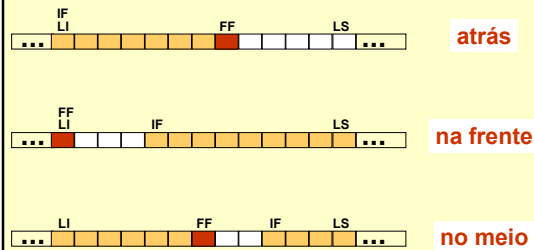
Operações válidas:

- criar uma fila vazia
- inserir um nó no final da fila
- excluir o nó do início da fila
- consultar / modificar nó do início da fila
- destruir a fila

Estruturas de Dados - Pilhas & Filas

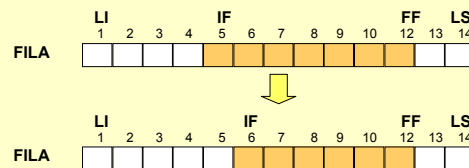
Inserção em fila - contigüidade física

Testar se tem espaço livre para inserir:



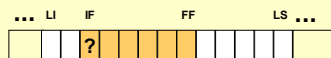
Estruturas de Dados - Pilhas & Filas

Remoção de fila - contigüidade física



Estruturas de Dados - Pilhas & Filas

Consulta à fila - contigüidade física

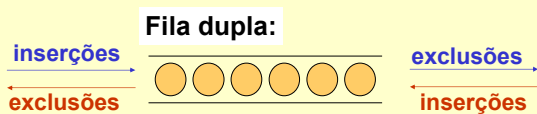


Estruturas de Dados - Pilhas & Filas

Dequeues Fila Dupla

Estruturas de Dados - Pilhas & Filas

Filas Duplas

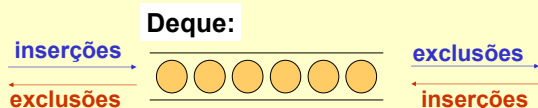


Inserções e exclusões podem ocorrer em qualquer extremidade da lista

- Especialização de fila

Estruturas de Dados - Pilhas & Filas

Operações sobre Deques



Operações válidas:

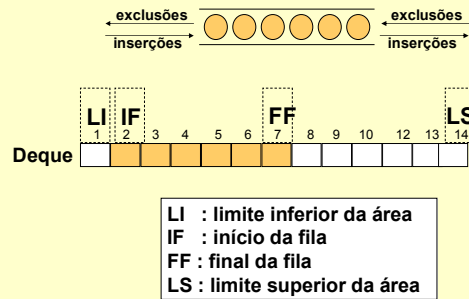
- criar uma fila dupla vazia
- inserir um nodo no início
- inserir um nodo no fim
- excluir um nodo do início
- excluir um nodo do fim
- consultar / modificar nodo do início ou do fim
- destruir a fila dupla

Estruturas de Dados - Pilhas & Filas

Filas Duplas Contigüidade física

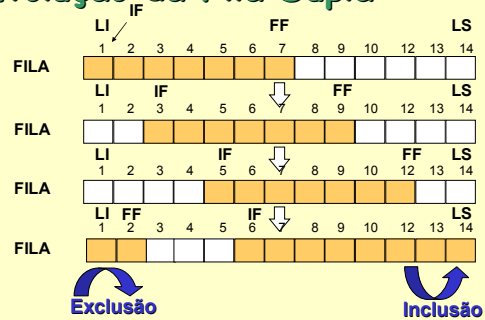
Estruturas de Dados - Pilhas & Filas

Fila Dupla implementado sobre arranjo



Estruturas de Dados - Pilhas & Filas

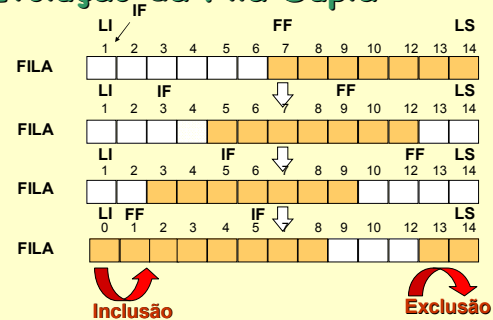
Evolução da Fila Dupla



Deslocamento Circular A DIREITA

Estruturas de Dados - Pilhas & Filas

Evolução da Fila Dupla



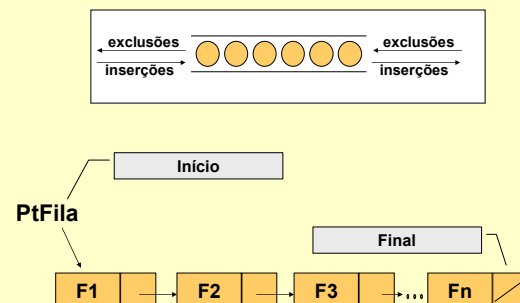
Deslocamento Circular A ESQUERDA
NOVIDADE!!!!!!

Estruturas de Dados - Pilhas & Filas

Filas Duplas Encadeamento

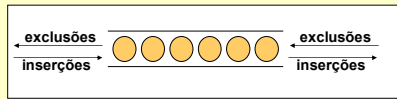
Estruturas de Dados - Pilhas & Filas

Filas Duplas - Encadeamento



Estruturas de Dados - Pilhas & Filas

Filas Duplas - Encadeamento



Descritor

tipo *DescrFila*

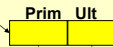
Prim: primeiro da fila

Ult : último da fila

Problema?????

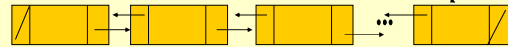
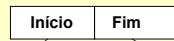
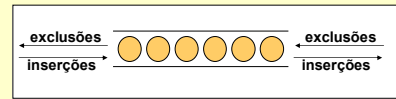


PtDF



Estruturas de Dados - Pilhas & Filas

Filas Duplas - Duplo Encadeamento



Estruturas de Dados - Pilhas & Filas