

Filas

Aula 05

Ivone P. Matsuno Yugoshi

`ivone.matsuno@ufms.br`

Ronaldo Fiorilo dos Santos

`ronaldo.santos@ufms.br`

Universidade Federal de Mato Grosso do Sul
Câmpus de Três Lagoas
Bacharelado em Sistemas de Informação

Algoritmos e Programação II

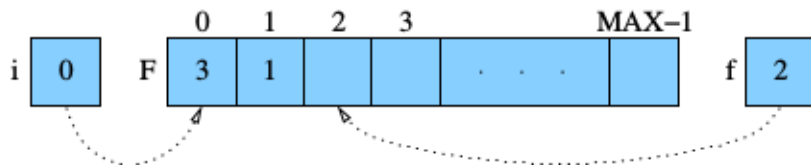
Slides baseados no material do Prof. Fábio Henrique Viduani Martinez - FACOM/UFMS

- ▶ **Fila** é uma lista linear com dois extremos destacados e tal que as operações de inserção são realizadas em um dos extremos da lista e a remoção é realizada no outro extremo
 - ▶ baseada no princípio FIFO (*First In, First Out*)
- ▶ O funcionamento dessa estrutura pode ser comparado a qualquer fila que usamos com frequência como, por exemplo, uma fila de um banco

Operações básicas em alocação sequencial

- ▶ A implementação mais simples de uma fila em alocação sequencial é armazenada em um segmento $F[i..f-1]$ de um vetor $F[0..MAX-1]$, com $0 \leq i \leq f \leq MAX$
- ▶ Onde o primeiro elemento da fila está na posição i e o último na posição $f-1$

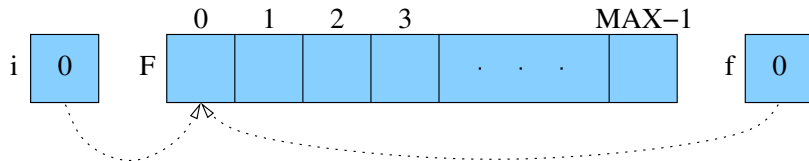
Definição



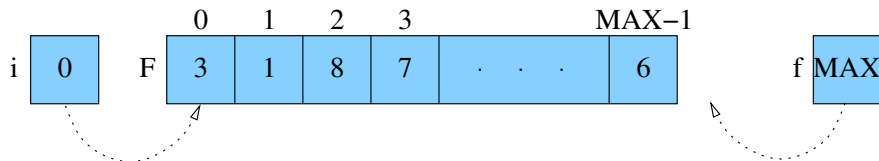
Operações básicas em alocação sequencial

- ▶ Convecionamos que uma fila está **vazia** se seus marcadores de início e fim são tais que $i = f$
- ▶ e dizemos que a fila está **cheia** se seu marcador de fim f vale **MAX**

Operações básicas em alocação sequencial



fila vazia



fila cheia

Operações básicas em alocação sequencial

- ▶ declaração e inicialização de uma fila em alocação sequencial

```
int i, f, F[MAX];  
i = 0;  
f = 0;
```

Operações básicas em alocação sequencial

- ▶ A operação de inserir um objeto em uma fila, ou **enfileirar**, é descrita na função `enfileira_seq` a seguir.

```
/* Recebe o índice f do fim da fila F e a chave y
e insere y no fim de F */
void enfileira_seq(int *f, int F[MAX], int y)
{
    if (*f != MAX) {
        F[*f] = y;
        (*f)++;
    }
    else
        printf("Fila cheia!\n");
}
```


Operações básicas em alocação sequencial

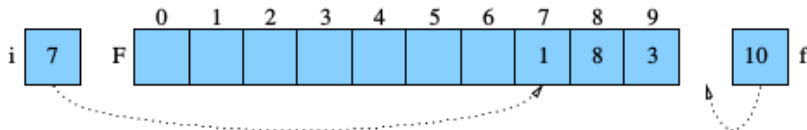
- ▶ A operação de remover, ou **desenfileirar**, um objeto de uma fila é descrita na função `desenfileira_seq` a seguir.

```
/* Recebe os índices i e f da fila F e remove a
chave do início i de F */
int desenfileira_seq(int *i, int f, int F[MAX])
{
    int x;

    if (*i != f) {
        x = F[*i];
        (*i)++;
    }
    else {
        x = INT_MAX;
        printf("Fila vazia!\n");
    }
    return x;
}
```

Operações básicas em alocação sequencial

► Problema!



Fila falsamente cheia

Operações básicas em alocação sequencial

- ▶ Considere que as células são alocadas sequencialmente como se estivessem em um círculo

Operações básicas em alocação sequencial

- ▶ Considere que as células são alocadas sequencialmente como se estivessem em um círculo
- ▶ O compartimento $F[\text{MAX}-1]$ é seguido pelo compartimento $F[0]$

Operações básicas em alocação sequencial

- ▶ Considere que as células são alocadas sequencialmente como se estivessem em um círculo
- ▶ O compartimento $F[\text{MAX}-1]$ é seguido pelo compartimento $F[0]$
- ▶ Assim, os elementos da fila estão dispostos no vetor $F[0.. \text{MAX}-1]$ em $F[i..f-1]$ ou em $F[i.. \text{MAX}-1]F[0..f-1]$

Operações básicas em alocação sequencial

- ▶ Considere que as células são alocadas sequencialmente como se estivessem em um círculo
- ▶ O compartimento $F[MAX-1]$ é seguido pelo compartimento $F[0]$
- ▶ Assim, os elementos da fila estão dispostos no vetor $F[0..MAX-1]$ em $F[i..f-1]$ ou em $F[i..MAX-1]F[0..f-1]$
- ▶ Temos também que $0 \leq i < MAX$ e $0 \leq f \leq MAX$.
- ▶ Com isso, dizemos que uma fila está **vazia**

Operações básicas em alocação sequencial

- ▶ Considere que as células são alocadas sequencialmente como se estivessem em um círculo
- ▶ O compartimento $F[MAX-1]$ é seguido pelo compartimento $F[0]$
- ▶ Assim, os elementos da fila estão dispostos no vetor $F[0..MAX-1]$ em $F[i..f-1]$ ou em $F[i..MAX-1]F[0..f-1]$
- ▶ Temos também que $0 \leq i < MAX$ e $0 \leq f \leq MAX$.
- ▶ Com isso, dizemos que uma fila está **vazia** se $i = f = -1$
- ▶ E a fila está **cheia**

Operações básicas em alocação sequencial

- ▶ Considere que as células são alocadas sequencialmente como se estivessem em um círculo
- ▶ O compartimento $F[MAX-1]$ é seguido pelo compartimento $F[0]$
- ▶ Assim, os elementos da fila estão dispostos no vetor $F[0..MAX-1]$ em $F[i..f-1]$ ou em $F[i..MAX-1]F[0..f-1]$
- ▶ Temos também que $0 \leq i < MAX$ e $0 \leq f \leq MAX$.
- ▶ Com isso, dizemos que uma fila está **vazia** se $i = f = -1$
- ▶ E a fila está **cheia** se $f = i \neq -1$

Operações básicas em alocação sequencial

- ▶ A operação de inserir um objeto em uma fila circular é descrita na função `enfileira_seq_2` a seguir.

```
/* Recebe os índices i e f da fila F e a chave y
e insere y no fim de F */
void enfileira_seq_2(int *i, int *f, int F[MAX], int y)
{
    if (*f != *i || *f == -1) {
        if (*f == -1) {
            *i = 0;
            *f = 0;
        }
        F[*f] = y;
        *f = (*f + 1) % MAX;
    }
    else
        printf("Fila cheia!\n");
}
```

Operações básicas em alocação sequencial

- ▶ A operação de remover um objeto em uma fila circular é descrita na função `desenfileira_seq_2` a seguir.

```
/* Recebe os índices i e f da fila F
e remove a chave do início de F */
int desenfileira_seq_2(int *i, int *f, int F[MAX])
{
    int r = INT_MIN;
    if (*i != -1) {
        r = F[*i];
        *i = (*i + 1) % MAX;
        if (*i == *f) {
            *i = -1;
            *f = -1;
        }
    }
    else
        printf("Fila vazia!\n");
    return r;
}
```

1. Escreva uma função que devolva o comprimento (ou seja, o número de elementos que a fila contém) de uma fila circular.
2. Implemente uma fila usando duas pilhas.
3. Implemente uma pilha usando duas filas.