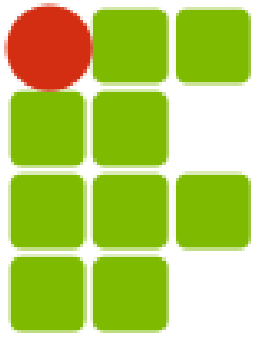


INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

Estrutura de Dados

Ricardo José Cabeça de Souza
www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

Parte 11

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - A ideia fundamental da fila é que só podemos inserir um novo elemento no final da fila e só podemos retirar o elemento do início
  - O primeiro que entra é o primeiro que sai" (a sigla **FIFO – *first in, first out*** – é usada para descrever essa estratégia)
  - Quem primeiro entra numa fila é o primeiro a ser atendido (a sair da fila)

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

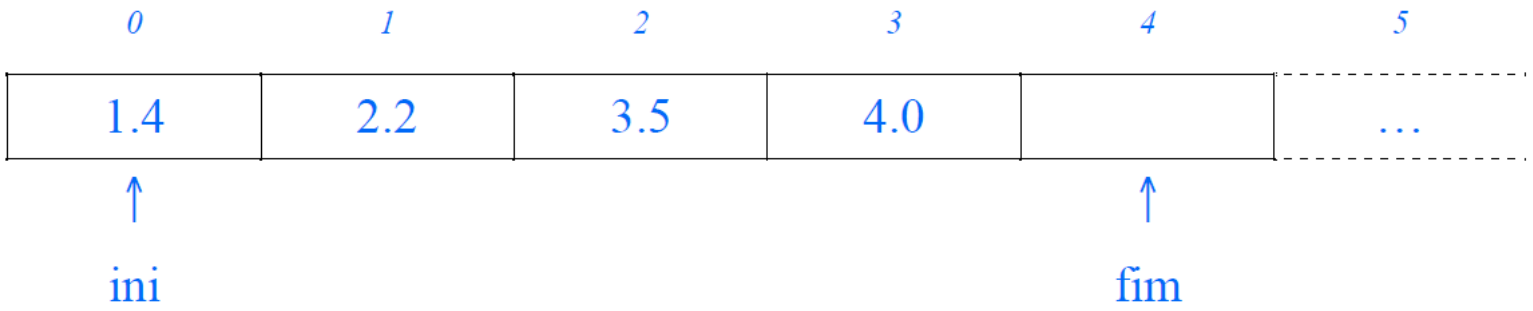
- **FILAS**
  - Operações:
    - criar uma estrutura de fila
    - inserir um elemento no fim
    - retirar o elemento do início
    - verificar se a fila está vazia
    - exibir os elementos da fila
    - liberar a fila

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - Criação da fila e alocação dinâmica da estrutura e inicializa a fila como sendo vazia
  - Os índices **ini** e **fim** iguais entre si (no caso, usamos o valor zero)



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**

```c
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#define N 100
/*Definição da estrutura da fila*/
struct estfila {
int ini, fim;
float vet[N];
};
typedef struct estfila fila;
/*Função cria: cria uma fila vazia*/
fila *cria (void)
{
fila* f = (fila*) malloc(sizeof(fila));
f->ini = 0; /* inicializa fila vazia */
f->fim = 0; /* inicializa fila vazia */
printf("\nFila criada com sucesso!");
return f;
}
main()
{
fila *f;
f = cria();
getch();
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
- Função de incremento
  - Função **incr** incrementa o valor em mais uma unidade
  - Função auxiliar responsável por incrementar o valor de um índice
  - Recebe o valor do índice atual e fornece com valor de retorno o índice incrementado

```
#define N 100
/*Função incr: incrementa o valor recebido em +1*/
int incr (int i)
{
if (i == N-1)
return 0;
else
return i+1;
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - Para inserir um elemento na fila, usamos a próxima posição livre do vetor, indicada por **fim**
  - Devemos ainda assegurar que há espaço para a inserção do novo elemento, tendo em vista que trata-se de um vetor com capacidade **limitada**

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**

Insere um elemento

```
/*Função insere: insere um elemento na fila*/
void insere (fila *f, float v)
{
if (incr(f->fim) == f->ini) { // fila cheia: capacidade esgotada
 printf("Capacidade da fila estourou.\n");
 exit(1); /* aborta programa */
}
/* insere elemento na próxima posição livre */
f->vet[f->fim] = v;
f->fim = incr(f->fim);
}
/*Função principal*/
main()
{
fila *f;
float valor;
f = cria();
printf("\nDigite um valor:");
scanf("%f",&valor);
insere(f,valor);
getch();
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**

Insere três elementos

```
/*Função insere: insere um elemento na fila*/
void insere (fila *f, float v)
{
if (incr(f->fim) == f->ini) { // fila cheia: capacidade esgotada
 printf("Capacidade da fila estourou.\n");
 exit(1); /* aborta programa */
}
/* insere elemento na próxima posição livre */
f->vet[f->fim] = v;
f->fim = incr(f->fim);
}
/*Função exibir: mostra os elementos da fila*/
void exibir (fila *fi)
{
int i;
for(i=fi->ini;i<fi->fim;i++)
 printf(" %.2f ",fi->vet[i]);
}
/*Função principal*/
main()
{
fila *f;
float valor;
int cont=0;
f = cria();
do {
printf("\nDigite um valor:");
scanf("%f",&valor);
insere(f,valor);
cont++;
}while(cont<3);
exibir(f);
getch();
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - Para exibir os elementos da fila

```
/*Função exibir: mostra os elementos da fila*/
void exibir (fila *fi)
{
int i;
for(i=fi->ini;i<fi->fim;i++)
 printf(" %.2f ",fi->vet[i]);
}
/*Função principal*/
main()
{
fila *f;
float valor;
int cont=0;
f = cria();
do {
printf("\nDigite um valor:");
scanf("%f",&valor);
insere(f,valor);
cont++;
}while(cont<3);
exibir(f);
getch();
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - Função que verifica se a fila está vazia

```c
/*Função vazia: verifica se a fila está vazia*/
int vazia (fila *f)
{
if (f->ini == f->fim)
return 0;
else
return 1;
}
```

```c
/*Função principal*/
main()
{
fila *f;
float valor;
int cont=0,op;
f = cria();
printf("\nPrimeira verificação");
if(vazia(f) == 0)
 printf("\n\nLista vazia!");
 else
 printf("\n\nLista nao esta vazia!");
do {
printf("\nDigite um valor:");
scanf("%f",&valor);
insere(f,valor);
cont++;
}while(cont<3);
exibir(f);
printf("\nSegunda verificação");
if(vazia(f) == 0)
 printf("\n\nLista vazia!");
 else
 printf("\n\nLista nao esta vazia!");
getch();
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - Função para retirar o elemento do início da fila
  - Fornece o valor do elemento retirado como retorno
  - Podemos também verificar se a fila está ou não vazia

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**

```c
/*Função retira: retira o primeiro elemento da fila*/
float retira (fila *f)
{
float v;
if (vazia(f)== 0) {
printf("Fila vazia.\n");
exit(1); /* aborta programa */
}
/* retira elemento do início */
v = f->vet[f->ini];
f->ini = incr(f->ini);
return v;
}
```

```c
/*Função principal*/
main()
{
fila *f;
float valor;
int cont=0,op;
f = cria();
printf("\nPrimeira verificação");
if(vazia(f) == 0)
 printf("\n\nLista vazia!");
 else
 printf("\n\nLista nao esta vazia!");
do {
printf("\nDigite um valor:");
scanf("%f",&valor);
insere(f,valor);
cont++;
}while(cont<3);
exibir(f);
printf("\nSegunda verificação");
if(vazia(f) == 0)
 printf("\n\nLista vazia!");
 else
 printf("\n\nLista nao esta vazia!");
printf("\nO elemento %.2f foi retirado da lista",retira(f));
printf("\nLista atualizada\n");
exibir(f);
getch();
}
```

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARÁ

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

FILAS

- **FILAS**
  - Função para liberar a memória alocada pela fila

```
/*Função libera: desaloca espaço de memória da fila*/
void libera (fila *f)
{
free(f);
}
/*Função principal*/
main()
{
fila *f;
float valor;
int cont=0,op;
f = cria();
printf("\nPrimeira verificação");
if(vazia(f) == 0)
 printf("\n\nLista vazia!");
 else
 printf("\n\nLista nao esta vazia!");
do {
printf("\nDigite um valor:");
scanf("%f",&valor);
insere(f,valor);
cont++;
}while(cont<3);
exibir(f);
printf("\nSegunda verificação");
if(vazia(f) == 0)
 printf("\n\nLista vazia!");
 else
 printf("\n\nLista nao esta vazia!");
printf("\nO elemento %.2f foi retirado da lista",retira(f));
printf("\nLista atualizada\n");
exibir(f);
libera(f);
getch();
}
```

www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARÁ

Estrutura de Dados

- **REFERÊNCIAS**
- Feofiloff, Paulo. **Projeto de Algoritmos em C**. Disponível em http://www.ime.usp.br/~pf/algoritmos/aulas/lista.html acesso em 12/07/2011.
- Tenenbaum, Aaron M. Langsam, Yedidyah, Augenstein, Moshe J. **Estruturas de dados usando C**. São Paulo : MAKRON *Books,* 1995.
- Veloso, Paulo. et. al. **Estrutura de dados**. Rio de Janeiro: Campus, 1986.
- Moraes, Celso Roberto. **Estrutura de dados e algoritmos**. 2. ed. São Paulo: Futura, 2003.
- Celes, W. Rangel, J. L. **Curso de Estrutura de Dados**. PUC-Rio, 2002.
- W. Celes, R. Cerqueira, J.L. Rangel. **Introdução a Estruturas de Dados - com técnicas de programação em C**. Rio de Janeiro: Campus, 2004.