



# INSTITUTO FEDERAL

## Ceará

Nome: Ana Vitória Cabral Duarte  
Matrícula: 20212013020205

Avaliação 08:

Implementar uma Fila usando uma Estrutura Encadeada com um ponteiro que aponta para o primeiro elemento da Fila e o ultimo elemento da Fila aponta para null.

Criar as funcionalidades :

- 1) Inserir um elemento na Fila;
- 2) Retirar o primeiro elemento da Fila;
- 3) Percorrer a Fila do inicio até o fim mostrando todos os elementos.

Printar a execução do programa para evidenciar a realização da tarefa e atualizar o github.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

// Definindo a estrutura para um nó da fila
struct Node {
    int data;
    struct Node* next;
};

// Definindo a estrutura da Fila
struct Queue {
    struct Node* front; // Ponteiro para o primeiro elemento
    struct Node* rear; // Ponteiro para o último elemento
};

// Função para criar uma fila vazia
struct Queue* createQueue() {
    struct Queue* queue = (struct Queue*)malloc(sizeof(struct Queue));
    queue->front = queue->rear = NULL;
    return queue;
}

// Função para inserir um elemento na fila
void enqueue(struct Queue* queue, int item) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = item;
```

```

newNode->next = NULL;

if (queue->rear == NULL) {
    queue->front = queue->rear = newNode;
    return;
}

queue->rear->next = newNode;
queue->rear = newNode;
}

// Função para retirar o primeiro elemento da fila
void dequeue(struct Queue* queue) {
    if (queue->front == NULL)
        return;

    struct Node* temp = queue->front;
    queue->front = temp->next;

    free(temp);
}

// Função para percorrer e mostrar todos os elementos da fila
void displayQueue(struct Queue* queue) {
    struct Node* current = queue->front;

    if (current == NULL) {
        printf("A fila está vazia.\n");
        return;
    }

    printf("Elementos na fila: ");
    while (current != NULL) {
        printf("%d ", current->data);
        current = current->next;
    }
    printf("\n");
}

int main() {
    setlocale(LC_ALL, "portuguese");

    struct Queue* fila = createQueue();

    enqueue(fila, 10);
    enqueue(fila, 20);
    enqueue(fila, 30);

    displayQueue(fila);

    dequeue(fila);
    displayQueue(fila);
}

```

```
    return 0;
}
```

The screenshot shows a C++ IDE with the following components:

- Editor:** Displays the source code for `avaliacao-08.cpp`. The code implements a queue using a linked list structure. It includes functions for creating the queue, enqueueing elements (10, 20, 30), displaying the queue, and dequeueing elements. The `main` function sets the locale to Portuguese and calls these functions.
- Output Window:** Shows the execution output. It prints "Elementos na fila: 10 20 30" and "Elementos na fila: 20 30". It also displays a message indicating the process exited after 0.05023 seconds with a return value of 0.
- Compiler Output Window:** Shows the compilation results. It reports 0 errors and 0 warnings. The output filename is `C:\Users\Ana Vitória\Documents\Linguagem de programação\avaliacao-08.exe`, the output size is 130,102,539,0625 KiB, and the compilation time is 1.06s.

```
58
59     printf("Elementos na fila: ");
60     while (current != NULL) {
61         printf("%d ", current->data);
62         current = current->next;
63     }
64     printf("\n");
65 }
66
67 int main() {
68     setlocale(LC_ALL, "portuguese");
69
70     struct Queue* fila = createQueue();
71
72     enqueue(fila, 10);
73     enqueue(fila, 20);
74     enqueue(fila, 30);
75
76     displayQueue(fila);
77
78     dequeue(fila);
79     displayQueue(fila);
80 }
```

Output:

```
Elementos na fila: 10 20 30
Elementos na fila: 20 30

-----
Process exited after 0.05023 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

Compilation results...

```
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Ana Vitória\Documents\Linguagem de programação\avaliacao-08.exe
- Output Size: 130,102,539,0625 KiB
- Compilation Time: 1,06s
```