

PROFESSOR: Dr Marcos Camada
ESTUDANTE: Oscar Borges

```
#include <stdio.h>
#include <stdlib.h>
#define tam 10

struct Fila {
    char dados [tam];
    int inicio, final, qtd;
};

struct Fila* criarFila() {
    struct Fila *fila;
    fila = (struct Fila*)malloc(sizeof(struct Fila));
    if(fila != NULL) {
        fila->qtd = 0;
        fila->inicio = 0;
        fila->final = 0;
    }
    return fila;
}

void excluirFila(struct Fila *fila) {
    free(fila);
}

int tamanhoFila(struct Fila *fila) {
    if(fila == NULL) return -1;
    else return fila->qtd;
}

int filaCheia(struct Fila *fila) {
    if(fila == NULL) return -1;
    else return (fila->qtd == tam);
}

int filaVazia(struct Fila *fila) {
    if(fila == NULL) return -1;
    else return (fila->qtd == 0);
}

int push(struct Fila *fila, char dado) {
```

```

    if(fila == NULL) {
        printf("Fila não existente\n");
        return 0;
    };
    if(filaCheia(fila)) {
        printf("Impossível adicionar '%c'. Fila Cheia!\n", dado);
        return 0;
    };
    fila->dados[fila->final] = dado;
    fila->final = (fila->final+1)%tam;
    fila->qtd++;
    printf("Dado '%c' inserido!\n", dado);
    return 1;
}

int pop(struct Fila* fila) {
    if(fila == NULL || filaVazia(fila)) {
        printf("Fila não existe ou está vazia. Nada para ser
removido!\n");
        return 0;
    };
    printf("Dado '%c' removido!\n", fila->dados[fila->inicio]);
    fila->inicio = (fila->inicio+1)%tam;
    fila->qtd--;
    return 1;
}

void verInicio(struct Fila* fila) {
    if(filaVazia(fila)) printf("Fila vazia. Não existe inicio!\n");
    else printf("Inicio da fila é o: %c\n", fila->dados[fila->inicio]);
}

int main(int argc, char const *argv[])
{
    struct Fila *fila = criarFila();
    push(fila, 'a');
    verInicio(fila);
    push(fila, 'b');
    verInicio(fila);
    push(fila, 'c');
    verInicio(fila);
    push(fila, 'd');
    verInicio(fila);
}

```

[illegible]