

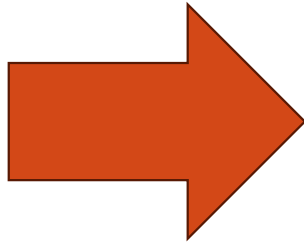
# INTEGRAÇÃO C/PYTHON



# INTEGRAÇÃO C/PYTHON

```
void enfileira(Fila* q, int valor) {
    Elemento* no = (Elemento*)malloc(sizeof(Elemento));
    no->dado = valor;
    no->prox = NULL;
    if (q->final == NULL) {
        q->inicio = q->final = no;
        return;
    }
    q->final->prox = no;
    q->final = no;
}

int desenfileira(Fila* q) {
    if (q->inicio == NULL) {
        return 0; // Queue is empty
    }
    Elemento* no = q->inicio;
    int valor = no->dado;
    q->inicio = q->inicio->prox;
    if (q->inicio == NULL) {
        q->final = NULL;
    }
    free(no);
    return valor;
}
```



gcc -shared -o libfila.so -fPIC fila.c



# INTEGRAÇÃO C/PYTHON

```
import ctypes

# Carregar a biblioteca compartilhada
lib = ctypes.CDLL('./libfila.so')

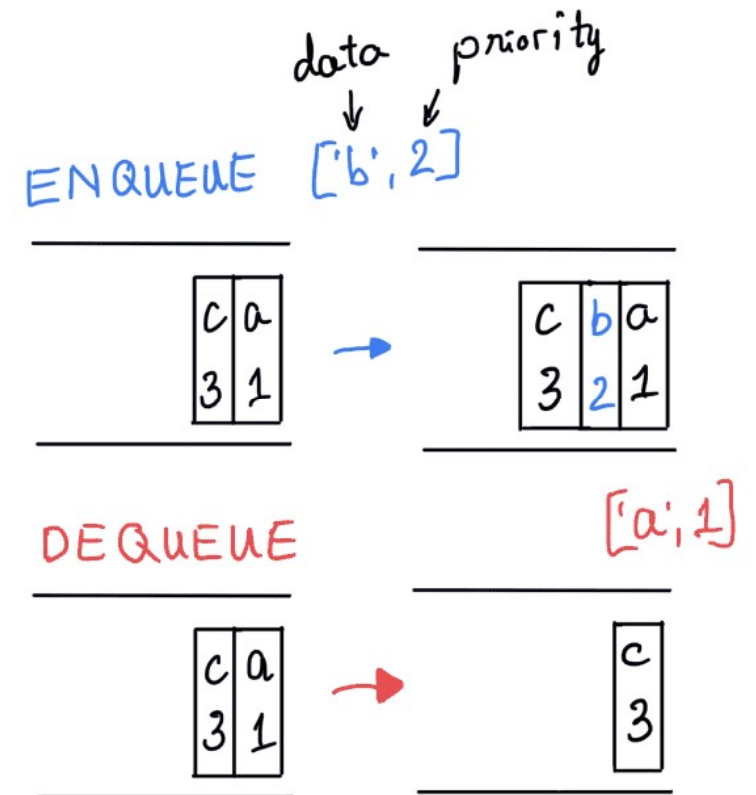
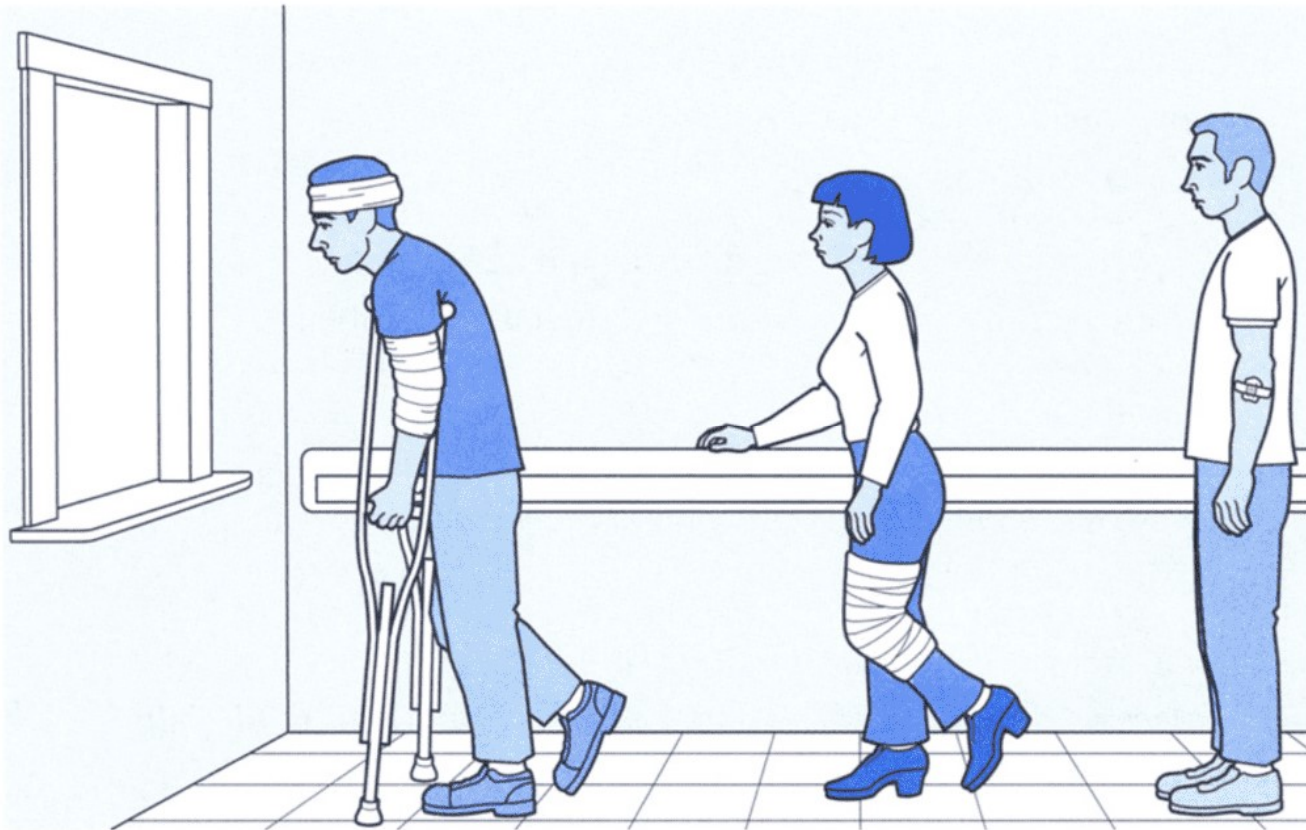
# Definir tipos de retorno e argumentos das funções
lib.criaFila.restype = ctypes.POINTER(ctypes.c_void_p)
lib.enfileira.argtypes = [ctypes.POINTER(ctypes.c_void_p), ctypes.c_int]
lib.desenfileira.argtypes = [ctypes.POINTER(ctypes.c_void_p)]
lib.desenfileira.restype = ctypes.c_int
lib.imprimeFila.argtypes = [ctypes.POINTER(ctypes.c_void_p)]

# Criar uma fila
fila = lib.criaFila()

lib.enfileira(fila, id)
valor = lib.desenfileira(fila)
```



# FILA DE PRIORIDADES





# FILA DE PRIORIDADES

---

- Tipo especial de fila onde os elementos inseridos possuem um campo extra associado a alguma prioridade.
- Essa prioridade determina a posição de inserção de um elemento na fila.

