

GRADUAÇÃO EM ANÁLISE E DESENVOLVIMENTO

PROGRAMAÇÃO DE COMPUTADORES II

Ponteiros / Alocação Dinâmica / Funções
(Trabalho 09)



ESTRUTURA DE DADOS: PILHA E FILA

1. Introdução

O trabalho proposto deve ser codificado na linguagem de programação “C”, tendo como objetivo trabalhar os conceitos referentes a ponteiros, alocação dinâmica, realocação dinâmica e funções lecionados em aula. Todos os requisitos especificados no documento devem ser cumpridos.

2. Conceito de Pilha

As pilhas são estruturas de dados baseadas no princípio **LIFO** (*last in, first out* – último a entrar, primeiro a sair), nesse contexto os dados inseridos por último numa pilha serão os primeiros a serem retirados. São utilizadas, por padrão, duas funções quando trabalhamos com pilhas: **PUSH**, responsável por inserir dados no topo da pilha, e **POP**, responsável por remover o dado contido no topo da pilha. A operação **POP** só ocorrerá se houverem dados na pilha, ou seja, caso ela não esteja vazia. Veja exemplo abaixo:

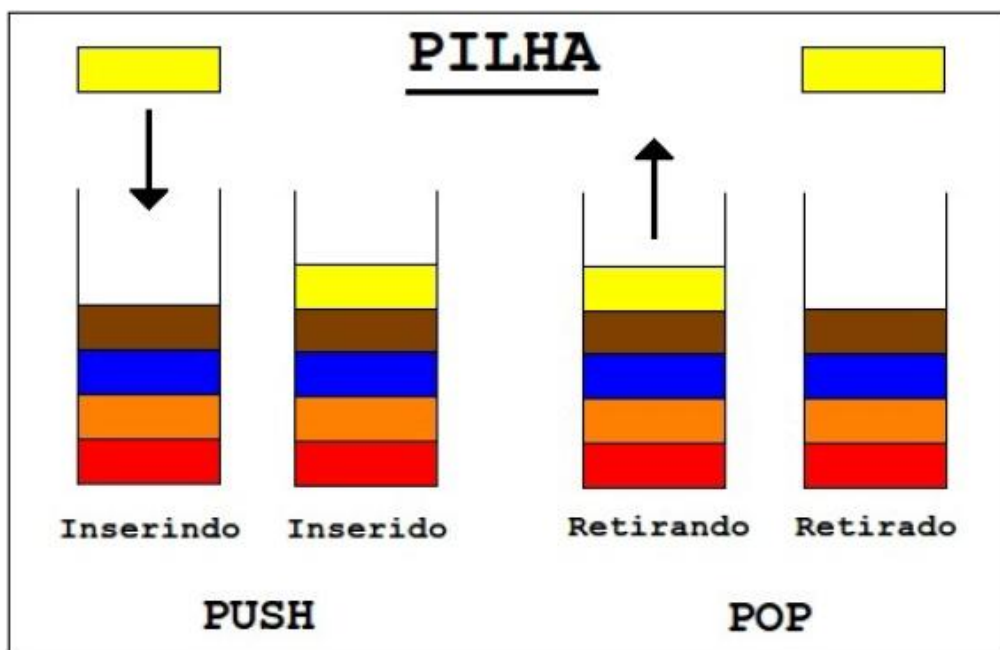


Figura 01: Inserindo e removendo dados de uma Pilha.

3. Conceito de Fila

As filas são estruturas de dados baseadas no princípio **FIFO** (*first in, first out* – primeiro a entrar, primeiro a sair), nesse contexto os dados inseridos por primeiro numa fila serão os primeiros a serem retirados. São utilizadas, por padrão, duas funções quando trabalhamos com filas: **ENQUEUE**, responsável por inserir (enfileirar) dados no final da fila, e **DEQUEUE**, responsável por remover o dado armazenado no início da fila. A operação **DEQUEUE** só ocorrerá se houverem dados na fila, ou seja, caso ela não esteja vazia. Veja exemplo abaixo:

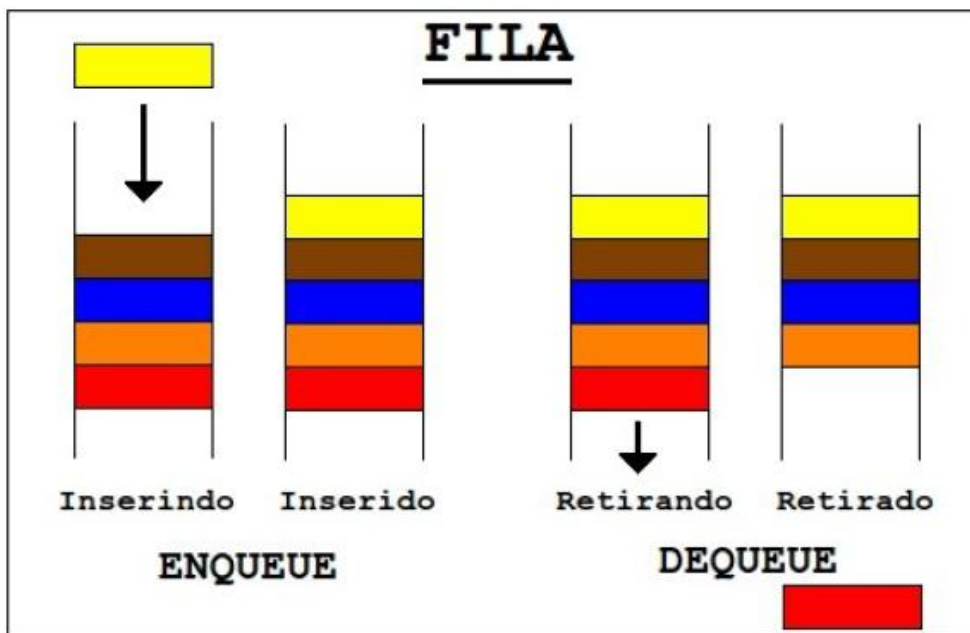


Figura 01: Inserindo e removendo dados de uma Fila.

4. Trabalho Proposto

Implemente uma pilha e uma fila para valores inteiros utilizando os conceitos de ponteiros e funções. A pilha e a fila não terão um tamanho pré-definido, ou seja, serão iniciadas com tamanho zero. A cada inserção efetuada na pilha ou na fila, os ponteiros utilizados para as representam serão realocados, incrementando o tamanho em "1" e armazenando o dado inteiro recebido. A cada remoção da pilha ou da fila, os ponteiros que as representam serão realocados, decrementando o tamanho em "1" e retirando o elemento em questão. Serão implementadas cinco funções:

- **int PUSH(int dado):** função que irá inserir o valor inteiro "dado" dentro da pilha. Ela deve retornar "1" caso a inserção tenha ocorrido corretamente e "-1" caso não seja possível inserir o valor.

➤

- **int POP():** função que removerá o valor inteiro armazenado no topo da pilha. Ela deve retornar o “**dado**” que está sendo removido da pilha caso a remoção tenha ocorrido corretamente, e “-1” caso não seja possível remover o valor.
- **int ENQUEUE(int dado):** função que irá inserir o valor inteiro “**dado**” dentro da fila. Ela deve retornar “1” caso a inserção tenha ocorrido corretamente e “-1” caso não seja possível inserir.
- **int DEQUEUE():** função que removerá o valor inteiro armazenado no final da fila. Ela deve retornar o “**dado**” que está sendo removido da fila caso a remoção tenha ocorrido corretamente, e “-1” caso não seja possível remover o valor.
- **void VIEW_LIFO():** função que permitirá visualizar o conteúdo da estrutura pilha.
- **void VIEW_FIFO():** função que permitirá visualizar o conteúdo da estrutura fila.

Toda a lógica necessária para construção da pilha e da fila deve ser codificada dentro de duas bibliotecas denominadas “pilha.h” e “fila.h”. Para que seja possível representar as estruturas “pilha.h” e “fila.h” devem ser utilizadas duas variáveis ponteiro do tipo *int*. Considere também que a pilha e a fila possuam um tamanho máximo de 10 elementos cada. O arquivo C principal (que possui a função *main()*) deve apenas obter (*scanf()*) os dados que serão armazenados nas estruturas e disponibilizar, ao usuário, um menu contendo as seguintes opções:

- ✓ Inserir dados na pilha
- ✓ Inserir dados na fila
- ✓ Remover dados da pilha
- ✓ Remover dado da fila
- ✓ Visualizar os dados da pilha
- ✓ Visualizar os dados da fila
- ✓ Sair

Os dados da pilha e da fila devem ser apresentados ao usuário de acordo com o seu formato de armazenamento, por exemplo:

PILHA	FILA
3 <i>(topo da pilha)</i> 2 1 <i>(base da pilha)</i>	1 2 3 <i>início da fila</i> <i>fim da fila</i>