

Pilhas

1. Crie um programa que implemente uma pilha utilizando um vetor fixo. Operações: push, pop, peek e isEmpty. Exiba o conteúdo da pilha a cada operação.
2. Implemente uma pilha utilizando alocação dinâmica de memória (ponteiros e malloc). Cada nó deve conter um número inteiro. Mostre o topo da pilha após cada inserção e remoção.
3. Usando uma pilha, crie um programa que leia uma expressão matemática (com (), {}, []) e verifique se os parênteses estão balanceados.
4. Escreva um programa que converta expressões da forma infixa para pós-fixa (notação polonesa reversa) usando uma pilha.
5. Simule um sistema de “Desfazer” de edições de texto. Cada comando digitado é empilhado. Ao pressionar “U” (Undo), o último comando é removido e mostrado.

Filas

6. Implemente uma fila utilizando vetor fixo. Operações: enqueue, dequeue, isEmpty, isFull. Exiba os elementos a cada operação.
7. Modifique a atividade anterior para que a fila funcione de forma circular, reaproveitando os espaços liberados.
8. Implemente uma fila usando lista encadeada (nós dinâmicos). Cada nó deve conter um número inteiro. Mostre o início e o fim da fila a cada operação.
9. Crie um programa que simule o atendimento em um caixa de banco. Cada cliente possui um número e tempo de atendimento. A fila deve ser atualizada conforme os clientes são atendidos.
10. Implemente uma fila em que cada elemento possui uma prioridade (inteiro). Elementos de maior prioridade devem ser atendidos antes. Use estrutura de fila dinâmica.