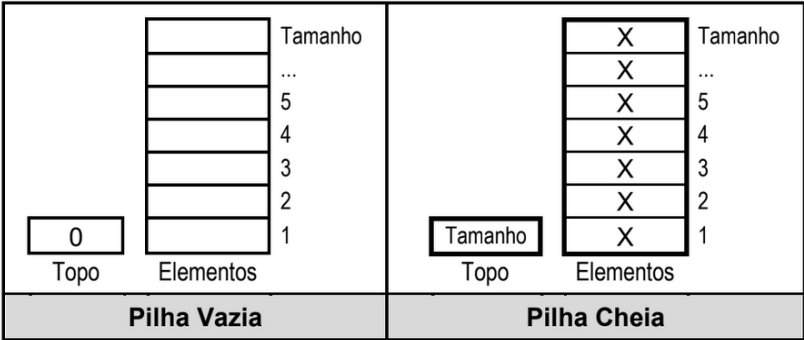
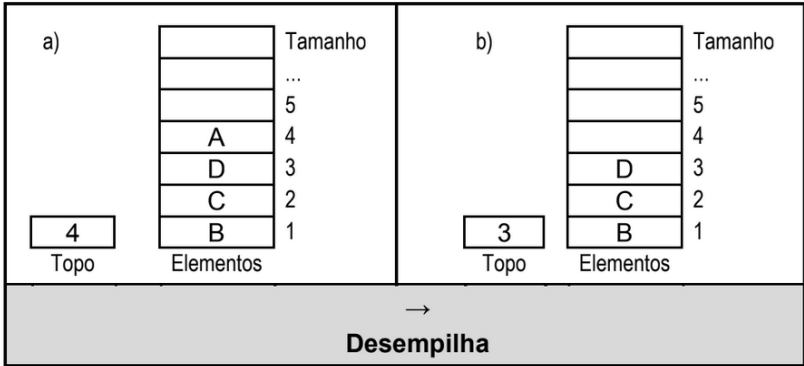
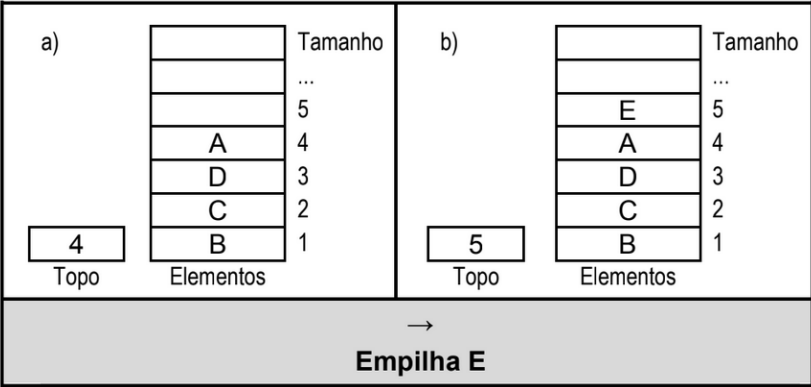
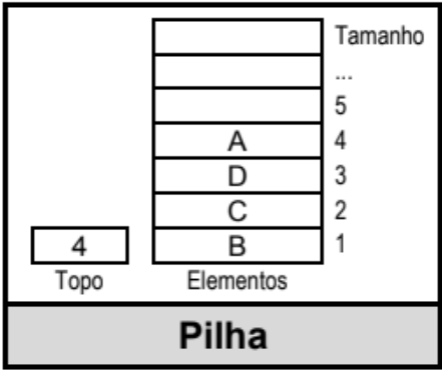


Diagramas da Pilha



## Código do arquivo pilha.h

```
#define TAMANHO 5 // Tamanho máximo da pilha

// Declaração da pilha
typedef struct Pilha {
    int elementos[TAMANHO];
    int topo;
} Pilha;

// Função que inicializa uma pilha vazia
void cria(Pilha *p) {
    p->topo = -1;
}

// Função que retorna 1 se a pilha está vazia e 0 caso contrário
int vazia(Pilha *p) {
    return p->topo == -1 ? 1 : 0;
}

// Função que retorna 1 se a pilha está cheia e 0 caso contrário
int cheia(Pilha *p) {
    return p->topo == TAMANHO - 1 ? 1 : 0;
}

// Função que adiciona um elemento à pilha caso ela não esteja cheia
void empilha(Pilha *p, int x, int *dev_certo) {
    if (cheia(p) == 1) {
        *dev_certo = 0;
    } else {
        p->topo++;
        p->elementos[p->topo] = x;
        *dev_certo = 1;
    }
}

// Função que remove o elemento no topo da pilha caso ela não esteja vazia
void desempilha(Pilha *p, int *x, int *dev_certo) {
    if (vazia(p) == 1) {
        *dev_certo = 0;
    } else {
        *x = p->elementos[p->topo];
        p->topo--;
        *dev_certo = 1;
    }
}

// Função que retorna o elemento no topo da pilha
int elemento_topo(Pilha *p, int *dev_certo) {
    if (vazia(p) == 1) {
        *dev_certo = 0;
    } else {
        int topo;

        desempilha(p, &topo, dev_certo);
        empilha(p, topo, dev_certo);

        *dev_certo = 1;

        return topo;
    }
}

// Função que retorna a quantidade total de elementos da pilha
int quantidade_elementos(Pilha *p, int *dev_certo) {
    if (vazia(p) == 1) {
        *dev_certo = 0;
    } else {
        *dev_certo = 1;
        return p->topo + 1;
    }
}
```

Imagem 1

```
#include "pilha.h" // Inclui a pilha

#include <stdio.h>

// Função que imprime os elementos atuais da pilha
void imprime(Pilha *p) {
    Pilha p_aux;
    int x;
    int ok;

    // Cria uma pilha auxiliar para realizar a impressões dos elementos
    cria(&p_aux);

    // Adiciona os elementos da pilha para a pilha auxiliar
    while (vazia(p) == 0) {
        desempilha(p, &x, &ok);

        if (ok) {
            empilha(&p_aux, x, &ok);
        }
    }

    printf("Imprimindo a pilha: ");

    // Imprime os elementos da pilha auxiliar enquanto até ela ficar vazia
    while (vazia(&p_aux) == 0) {
        desempilha(&p_aux, &x, &ok);

        if (ok) {
            printf("%d ", x);
            empilha(p, x, &ok);
        }
    }

    printf("← Topo\n");
}
```

## Imagem 2

```
// Função main que usa as funções da pilha
int main() {
    Pilha p;
    int ok;
    int a;

    cria(&p);

    printf("\n
Tentando empilhar 6 elementos em uma Pilha que so cabem 5\n");

    for (int i = 10; i ≤ 60; i += 10) {
        empilha(&p, i, &ok);
        imprime(&p);
    }

    printf("Pressione uma tecla... ");
    getchar();

    printf("\n
Tentando desempilhar 6 elementos em uma Pilha que so tem 5\n");

    int j = 1;
    while (j ≤ 6) {
        desempilha(&p, &a, &ok);
        imprime(&p);
        j++;
    }

    printf("Pressione uma tecla... ");
    getchar();

    printf("\nAdicionando os valores 5 e 15 à pilha\n");

    empilha(&p, 5, &ok);
    empilha(&p, 15, &ok);
    imprime(&p);

    int topo = elemento_topo(&p, &ok);
    if (ok) {
        printf("\nElemento no topo da pilha: %d\n", topo);
    }

    int total = quantidade_elementos(&p, &ok);
    if (ok) {
        printf("Total de elementos na pilha atual: %d\n", total);
    }

    return 0;
}
```

## Imagem da execução do programa

```
ranzani in AED1/Frequências/F2 on ʘ main [X!?] took 6s
→ gcc usa_pilha.c -o usa_pilha
ranzani in AED1/Frequências/F2 on ʘ main [X!?]
→ ./usa_pilha

Tentando empilhar 6 elementos em uma Pilha que so cabem 5
Imprimindo a pilha: 10 <- Topo
Imprimindo a pilha: 10 20 <- Topo
Imprimindo a pilha: 10 20 30 <- Topo
Imprimindo a pilha: 10 20 30 40 <- Topo
Imprimindo a pilha: 10 20 30 40 50 <- Topo
Imprimindo a pilha: 10 20 30 40 50 <- Topo
Pressione uma tecla...

Tentando desempilhar 6 elementos em uma Pilha que so tem 5
Imprimindo a pilha: 10 20 30 40 <- Topo
Imprimindo a pilha: 10 20 30 <- Topo
Imprimindo a pilha: 10 20 <- Topo
Imprimindo a pilha: 10 <- Topo
Imprimindo a pilha: <- Topo
Imprimindo a pilha: <- Topo
Pressione uma tecla...

Adicionando os valores 5 e 15 à pilha
Imprimindo a pilha: 5 15 <- Topo

Elemento no topo da pilha: 15
Total de elementos na pilha atual: 2
```