

# Conceitos de TAD aplicados na Pilha em C

## (Semana 13)

### 1. Abstração

A estrutura interna da pilha (vetor e índice do topo) não é visível ao usuário. O usuário interage apenas por funções, como push, pop, peek, etc.

No código: Em pilha.h, temos apenas: `typedef struct Pilha Pilha;` // oculta a estrutura real

### 2. Encapsulamento

O conteúdo interno da pilha (`dados[]` e `topo`) está apenas no `pilha.c`. Só é possível manipular a pilha por meio das funções públicas declaradas no `.h`.

Isso permite alterações internas sem impactar o programa principal (`main.c`).

### 3. Interface clara

A interface da pilha define operações bem delimitadas (push, pop, peek, etc). O usuário não precisa saber como essas funções funcionam, apenas o que fazem.

### 4. Modularização

O TAD Pilha é dividido corretamente: `pilha.h` (interface), `pilha.c` (implementação), `main.c` (cliente do TAD). Isso favorece a manutenção, organização e reutilização.

### 5. Instanciar estruturas

O código usa o `typedef struct Pilha` para definir o tipo abstrato. Funções são usadas para interagir com a estrutura de forma segura.