



UNINASSAU



Estrutura de Dados

Anderson Lima
Aula 2

Agenda

- Conceitos básicos de estrutura de dados
- Tipo abstrato de dados

Conceitos básicos

Conceitos básicos

Tipo de Dado

- Um tipo de dado define o conjunto de valores (domínio) e operações que uma variável pode assumir.
- Ex: int, char, float e double.

Estrutura de Dados

- Uma estrutura de dados consiste em um conjunto de tipos de dados em que existe algum tipo de relacionamento lógico estrutural.
- Ex: array, struct, union e enum.

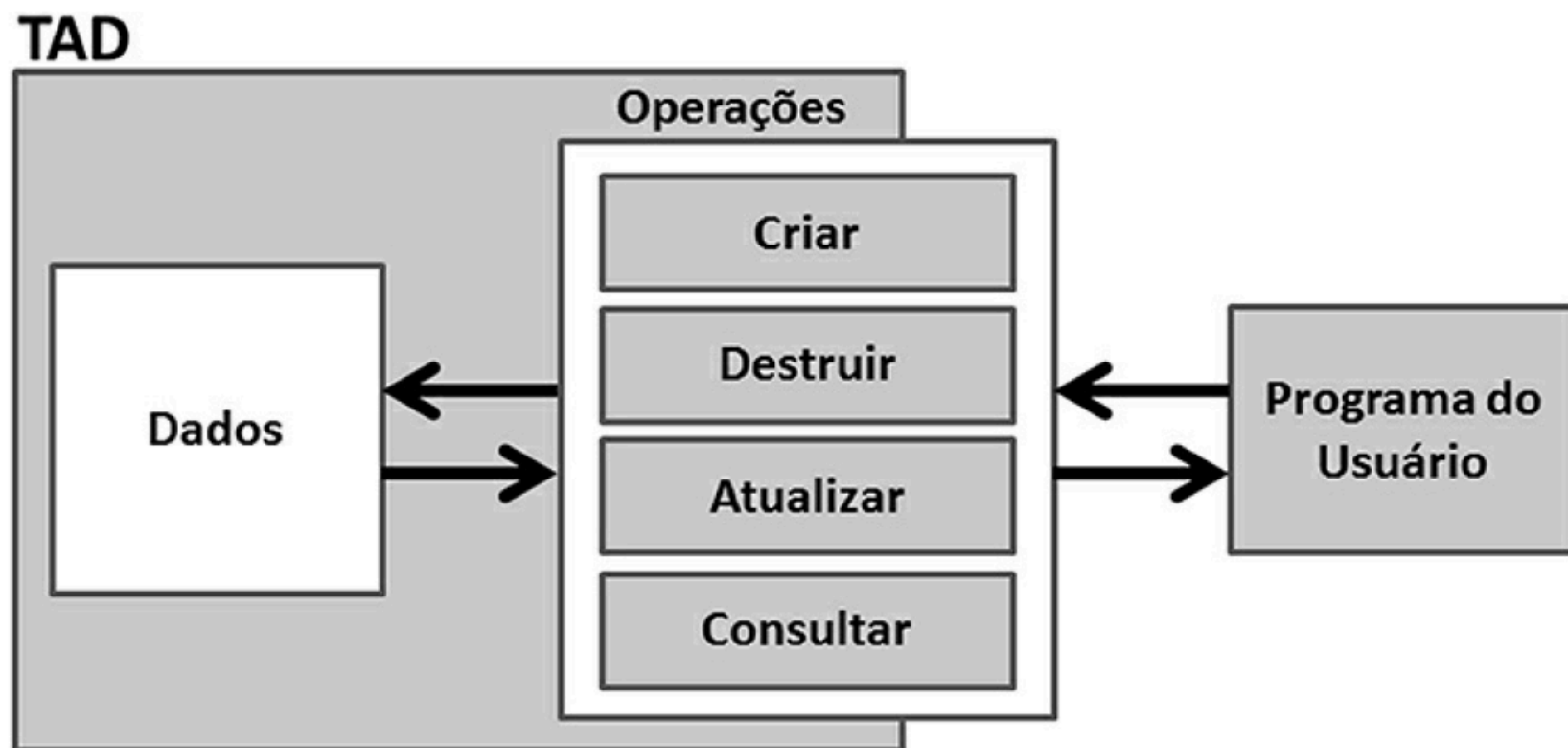
Conceitos básicos

Tipo abstrato de dados

- Um tipo abstrato de dados, ou TAD, é um conjunto de dados estruturados e as operações que podem ser executadas sobre esses dados.
- Um tipo abstrato de dados é um conjunto de valores com seu comportamento definido por operações implementadas na forma de funções.

Conceitos básicos

Tipo de dado abstrato



Conceitos básicos

Vantagens do TAD

- Encapsulamento
- Segurança
- Flexibilidade
- Reutilização

Conceitos básicos

Operações básicas do TAD

- Criação do TAD
- Inserção de um novo elemento no TAD
- Remoção de um elemento do TAD
- Acesso a um elemento do TAD
- Destruição do TAD

Conceitos básicos

Modularização

- Arquivo de definição (header) - .h
- Arquivo de implementação (corpo) - .c

Conceitos básicos

Exemplo de implementação

- TAD para representar um Ponto (x,y)

Conceitos básicos

Exemplo de implementação

Arquivo Ponto.h

```
01 typedef struct ponto Ponto;  
02 //Cria um novo ponto  
03 Ponto* Ponto_cria(float x, float y);  
04 //Libera um ponto  
05 void Ponto_libera(Ponto* p);  
06 //Acessa os valores "x" e "y" de um ponto  
07 int Ponto_acessa(Ponto* p, float* x, float* y);  
08 //Atribui os valores "x" e "y" a um ponto  
09 int Ponto_atribui(Ponto* p, float x, float y);  
10 //Calcula a distância entre dois pontos  
11 float Ponto_distancia(Ponto* p1, Ponto* p2);
```

Arquivo Ponto.c

```
01 #include <stdlib.h>  
02 #include <math.h>  
03 #include "Ponto.h" //inclui os Protótipos  
04 struct ponto{//Definição do tipo de dados  
05     float x;  
06     float y;  
07 };
```

Conceitos básicos

Criação de um ponto

Criando um ponto

```
01 Ponto* Ponto_cria(float x, float y) {  
02     Ponto* p = (Ponto*) malloc(sizeof(Ponto));  
03     if(p != NULL) {  
04         p->x = x;  
05         p->y = y;  
06     }  
07     return p;  
08 }
```

Conceitos básicos

Destruindo um ponto

Destruindo um ponto	
01	void Ponto_libera(Ponto* p) {
02	free(p);
03	}

Conceitos básicos

Acessando um ponto

Acessando o conteúdo de um ponto

```
01  int Ponto_acessa(Ponto* p, float* x, float* y){  
02      if(p == NULL)  
03          return 0;  
04      *x = p->x;  
05      *y = p->y;  
06      return 1;  
07  }
```


Conceitos básicos

Atribuindo o valor a um ponto

Atribuindo um valor ao ponto	
01	int Ponto_atribui(Ponto* p, float x, float y){
02	if (p == NULL)
03	return 0;
04	p->x = x;
05	p->y = y;
06	return 1;
07	}

Conceitos básicos

Calculando a distância entre dois pontos

Calculando a distância entre dois pontos	
01	float Ponto_distancia(Ponto* p1, Ponto* p2) {
02	if (p1 == NULL p2 == NULL)
03	return -1;
04	float dx = p1->x - p2->x;
05	float dy = p1->y - p2->y;
06	return sqrt(dx * dx + dy * dy);
07	}

Conceitos básicos

Exemplo de utilização do TAD

Exemplo: utilizando o TAD ponto

```
01  #include <stdio.h>
02  #include <stdlib.h>
03  #include "Ponto.h"
04  int main() {
05      float d;
06      Ponto *p,*q;
07      //Ponto r; //ERRO
08      p = Ponto_cria(10,21);
09      q = Ponto_cria(7,25);
10      //q->x = 2; //ERRO
11      d = Ponto_distancia(p,q);
12      printf("Distancia entre pontos: %f\n",d);
13      Ponto_libera(q);
14      Ponto_libera(p);
15      system("pause");
16      return 0;
17 }
```

Conceitos básicos

Exercícios

1. Desenvolva um TAD que represente um cubo. Inclua as funções de inicializações necessárias e as operações que retornem os tamanhos de cada lado, a sua área e o seu volume.
2. Desenvolva um TAD que represente uma esfera. Inclua as funções de inicializações necessárias e as operações que retornem o seu raio, a sua área e o seu volume.

Cenas do próximo capítulo...

O que nós vimos hoje?

- Conceitos básicos de estrutura de dados
- Tipos abstratos de dados (TAD)

Na próxima aula...

- Listas sequenciais

Anderson Lima

andclima@gmail.com

