

Estrutura de Dados I

aula 1

Professor: Manoel Messias Pereira Medeiros

Introdução a Estrutura de Dados (revisão de alguns conceitos)

Objetivo da disciplina

- **Familiarizar os alunos com a modelagem e a implementação de diferentes estruturas de dados, bem como os algoritmos para gerenciá-las.**
- **Durante a disciplina as implementações serão feitas com a linguagem de programação C, no entanto este não é o objetivo desta disciplina ensinar esta linguagem.**

Estrutura de dados

- Trata-se de um relacionamento lógico entre tipos de dados
 - Por exemplo um scruct da linguagem C
 - Uma **struct** é uma variável especial que contém diversas outras variáveis normalmente de tipos diferentes. As variáveis internas contidas pela **struct** são denominadas membros da **struct**. Podemos dizer que as **structs** da linguagem **C** são o equivalente ao que se denomina registros em outras linguagens de programação.

Ex:

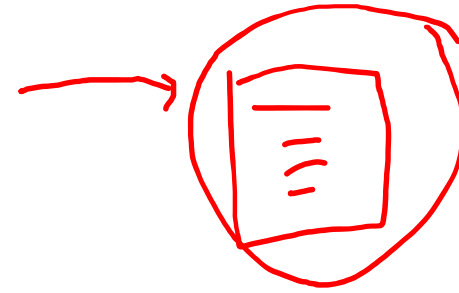
```
struct ficha_de_aluno  
{  
  char nome[50];  
  char disciplina[30];  
  float nota_prova1;  
  float nota_prova2;  
};
```

Heterogeneidade

Tipos Abstratos de Dados - TAD

- Tipo Abstratos de Dados ou “TAD” inclui as operações para manipulação dos dados.
- Ex.:
 - Criação da estrutura
 - Inclusão de um elemento
 - Remoção de um elemento
 - Acesso a um elemento
 - etc

Tipos Abstratos de Dados - TAD



- Vantagens:
- Encapsulamento e Segurança: usuário não tem acesso direto aos dados
- Flexibilidade e reutilização: podemos alterar o “TAD” sem alterar as aplicações que utilizam

Tipo de dados

- Tipos de dados define um conjunto de valores (domínio) que uma variável pode assumir.
- Tipos de dados da linguagem C
 - Existem 5 tipos de dados primitivos (pré-definidos) em C. São eles:

Palavra Chave	Tipo
char	caracter
int	inteiro
float	real de precisão simples
double	real de precisao dupla
void	vazio (sem valor)

boolean

Tipo de dados

- O tamanho e a faixa de valores depende do computador e do compilador utilizado.
- O padrão ANSI C determina somente um intervalo de valores mínimo para cada tipo de dado.

Modificadores de Tipos

- Com exceção de void, os outros tipos de dados primitivos podem ter modificadores.
- Os modificadores alteram o tamanho do tipo de dado ou sua forma de representação.
- Sua utilização faz com que seja possível adequar-se melhor às necessidades de armazenamento de dados em determinados casos.
- Veja quais são os modificadores na tabela abaixo.

Palavra Chave	Tipo
signed	caracter
unsigned	Inteiro
long	longo
short	curto

Tipo de dados

Tipos de dados e modificadores.

Palavra chave	Tipo	Tamanho	Intervalo
char	Caracter	1	-128 a 127
signed char	Caractere com sinal	1	-128 a 127
unsigned char	Caractere sem sinal	1	0 a 255
int	Inteiro	2	-32.768 a 32.767
signed int	Inteiro com sinal	2	-32.768 a 32.767
unsigned int	Inteiro sem sinal	2	0 a 65.535
short int	Inteiro curto	2	-32.768 a 32.767
signed short int	Inteiro curto com sinal	2	-32.768 a 32.767
unsigned short int	Inteiro curto sem sinal	2	0 a 65.535
long int	Inteiro long	4	-2.147.483.648 a 2.147.483.647
signed long int	Inteiro longo com sinal	4	-2.147.483.648 a 2.147.483.647
unsigned long int	Inteiro longo sem sinal	4	0 a 4.294.967.295
float	Ponto flutuante com precisão simples	4	<u>3.4 E-38 a 3.4E+38</u>
double	Ponto flutuante com precisão simples	8	1.7 E-308 a 1.7E+308
long double	Ponto flutuante com precisão dupla longo	16	3.4E-4932 a 1.1E+4932

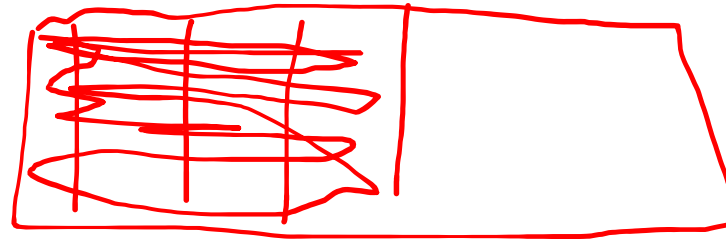


Linguagem fortemente tipadas

- Em linguagens fortemente tipadas, é preciso declarar o tipo.

Ex. de criação de variável na linguagem C

```
→ int num;  
   float nota1;
```



Variáveis indexadas Homogêneas

- Pode acontecer de armazenar, processar e, finalmente, imprimir dezenas, talvez centenas, talvez milhares de números. O que então? Você precisa criar uma variável separada para cada valor? Você terá que passar longas horas escrevendo declarações como essa aqui:
- `int var1=15`
- `int var2=20`
- `int var3=16`
- `int var4=25`
- `int var5=17`

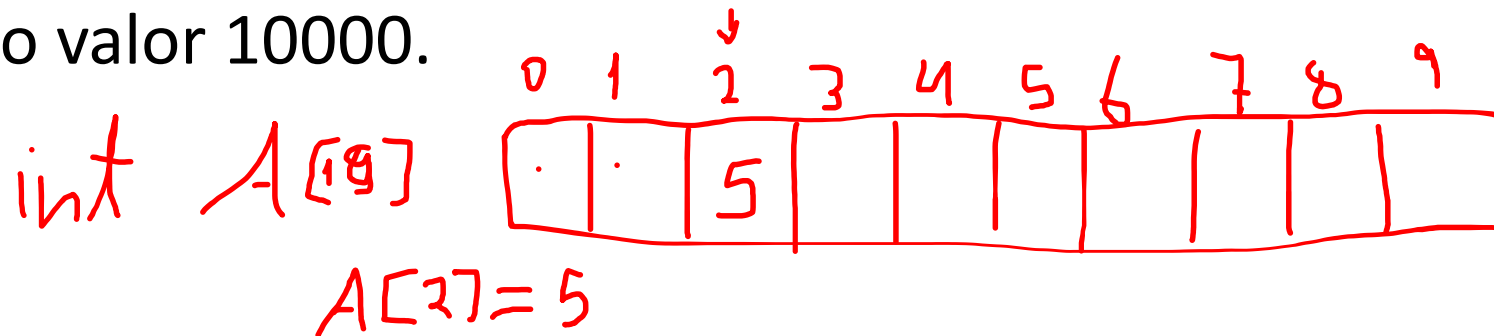
Variáveis indexadas Homogêneas

- Se você não acha que essa é uma tarefa complicada, pegue um pedaço de papel e escreva um programa que leia cinco números, imprima-os em ordem do menor para o maior (esse tipo de processamento é chamado de classificação). Você deve descobrir que nem sequer tem papel suficiente para concluir a tarefa.

Variáveis indexadas Homogêneas

- Pense em como seria conveniente declarar uma variável que pudesse armazenar mais de um valor. Por exemplo, 100 ou 1000 ou até 10.000. Ainda seria uma e a mesma variável, mas muito ampla e ampla. Soa atraente? Talvez, mas como lidaria com um contêiner tão cheio de valores diferentes? Como escolheria apenas o que você precisa?

E se você pudesse simplesmente numerá-los? E então diga: me dê o valor número 2; atribuir o número de valor 15; aumentar o número do valor 10000.



Variáveis indexadas Homogêneas


- A solução para esse problema é utilizar uma variável indexada. Também conhecidas como matrizes, que são conjunto de dados do mesmo tipo.
- Por exemplo imagine o problema de criar e gerenciar variáveis para guardar quatro notas de um aluno. Ao invés de criarmos uma variável para cada nota criamos uma única variável indexada homogênea para guardar essas informações.
- Nota

10.0	8.5	9.7	7.8
------	-----	-----	-----
- Uma matriz como a anterior é dita uma matriz de uma única dimensão ou vetor (pode ainda ser chamada de arranjo). (que pode ser de uma única linha ou de uma única coluna)

Variáveis indexadas Homogêneas

- Criando um vetor na linguagem C
- Por exemplo um vetor que guardará as quatro notas de um aluno como o exemplo anterior.

```
float Nota[4];
```

A hand-drawn red bracket is positioned below the array declaration 'float Nota[4];', spanning the width of the array name and its size, likely to highlight the array part of the code.

- A declaração acima cria um vetor com quatro posições do tipo float.

Variáveis indexadas Homogêneas

- Acessando e modificando os valores de um vetor.
- Para acessar um valor de um vetor é preciso se referenciar ao seu índice, por isso ele é chamado de variável indexada.
- Por exemplo código abaixo atribui o valor 10 a primeira posição do vetor notas.


notas[0]=1;

** Como em outras linguagens o numero de um vetor em C. começa com zero.*

Exercício

- Crie um vetor que leia as quatros notas de um aluno, calcule a média e imprima a média calculada.
- Dica: Você obviamente terá que usar uma estrutura de repetição, para controlar o acesso as dados de um vetor.