

Algoritmos e Estrutura de Dados II - 2º BSI

Lista de Exercícios 4

Aluno: Mateus Filipe de Lima Souza

1. A finalidade de definir estruturas é:

c) Criar novos tipos de dados;

2. Quantas variáveis de um tipo já definido por meio da palavra **struct** podem ser criadas em um programa?

c) Quantas forem necessárias;

3. Verdadeiro ou Falso: a sintaxe da declaração de uma variável de um tipo **struct** difere da sintaxe da declaração de uma variável de tipos simples.

➤ **Falso**

4. Para acessar um membro de uma variável **struct**, o operando à esquerda do ponto deve ser:

d) O nome da variável;

5. Suponha que as seguintes declarações tenham sido feitas:

```
struct Corpo
{
    float altura;
    float peso;
};

Corpo joao;
```

a)

```
struct Corpo{
    float altura;
    float peso;
}Joao;
```

b) joao.altura = 1.68;

c) struct casal {

 corpo homem;

 corpo mulher;

};

d) struct Corpo{
 float altura;
 float peso;
};
struct casal {

 corpo homem;

 corpo mulher;

};

int maint(){

 casal vet[10];

}

e) corpo maria = {1.63,56.5};

corpo joao = {1.78,82.6};

casal vet[10];

vet[0].homem = joao;

vet[0].mulher = maria;

6. Dada a seguinte instrução, quais alternativas são verdadeiras?

aaa.bbb.ccc = 25;

a) ccc é membro da estrutura bbb;

b) bbb é membro da estrutura aaa;

7. Verdadeiro ou Falso: é possível passar uma variável estrutura para uma função do mesmo modo que passamos uma variável simples.

➤ **Verdadeiro**

8. Escreva uma estrutura para conter três membros do tipo **int** chamados hora, minutos e segundos. Atribua o nome **tempo** a essa estrutura.

```
struct tempo {  
    int hora;  
    int minutos;  
    int segundos;  
};
```

9. Escreva uma estrutura para armazenar dados de um estacionamento. Ela deve ser capaz de armazenar o número da chapa do carro, a marca, a hora de entrada e a hora de saída do estacionamento. Utilize dois membros do tipo tempo, definido no exercício anterior, para as horas de entrada e saída.

```
struct estacionamento{  
    int chapa;  
    string marca;  
    tempo entrada;  
    tempo saida;  
};
```

10. Um tipo **enum** é definido para agrupar:

c) Inteiros com nomes definidos pelo programador;