

**Respostas Lista de Exercícios 01 sobre
Tipos Abstratos de Dados (TAD) e Estruturas de Dados em C**

**Sistemas de Informação - UFG
Aluno: Marcos Vinicius Vieira da Silva
Matrícula: 147255**

A maioria dos exercícios foram criados utilizando a IDE CLion e para cada um foi criado um projeto (pasta) com os arquivos .c e .h. que será enviada junto com esse PDF.

1 - **R:** pasta: exercício-1

2 - **R:** pasta: exercicio-2

3 - **R:**

Definições das Estruturas

A estrutura nametype tem os seguintes campos:

char first[10]: 10 bytes

char midinit: 1 byte

char last[20]: 20 bytes

Total: 31 bytes

A estrutura person tem os seguintes campos:

struct nametype name: 31 bytes

int birthday[2]: 2 inteiros = $2 \times 4 = 8$ bytes

struct nametype parentes[2]: 2 vezes nametype = $2 \times 31 = 62$ bytes

int income: 4 bytes

int numchildren: 4 bytes

char address[20]: 20 bytes

char city[10]: 10 bytes

char state[2]: 2 bytes

Total de person: 141 bytes

a) p[10]: 1510 bytes

b) p[20].name.midinit: 2930 bytes

c) p[20].income: 2990 bytes

d) p[20].address[5]: 3003 bytes

e) p[5].parentes[1].last[10]: 865 bytes

4 - **R:** pasta: exercicio-4

5 - **R:** pasta: exercicio-5

6 - R: pasta: exercicio-6

7 - R: pasta: exercicio-7

8 - R: pasta: exercicio-8

Preferência Entre os Métodos:

- **Método 1 (Simplificação)** é mais seguro e fácil de entender, pois sempre trabalha com os termos simplificados dos números racionais.
- **Método 2 (Multiplicação Cruzada)** pode ser mais eficiente em termos de tempo, já que evita o cálculo do MDC, mas apresenta risco de **overflow** em casos de números grandes.

Em geral, o **Método 1 (Simplificação)** é preferível, especialmente se os números racionais puderem ser simplificados antes de realizar outras operações.

9 - R:

```
C/C++
typedef struct {
    char nome[50];           // Nome (literal, uma string com até 50 caracteres)
    double salario;         // Salario (numerico, valor decimal)
    int idade;               // Idade (numerico, valor inteiro)
    bool sexo;               // Sexo (logico, true para masculino e false para feminino)
} REG;
```

10 - R: pasta: exercicio-10

11 - R: pasta: exercicio-11

12 - R: pasta: exercicio-12

13 - R: pasta: exercicio-13

14 - R: pasta: exercicio-14