

```

#include <stdio.h>
#include <string.h>

struct dados_pacientes
{
    int idade, e_civil;
    char nome[10];
    double peso, altura;
    float grau_miopia[2];
};

int main() {
    struct dados_pacientes paciente;

    strcpy(paciente.nome, "Jose");
    paciente.altura = 1.25;
    paciente.peso = 73;
    paciente.e_civil = 1;
    paciente.grau_miopia[0] = 1.75;
    paciente.grau_miopia[1] = 0;

    printf("Tamanho de idade: %lu bytes\n", sizeof(paciente.idade));
    printf("Tamanho de e_civil: %lu bytes\n", sizeof(paciente.e_civil));
    printf("Tamanho de nome: %lu bytes\n", sizeof(paciente.nome));
    printf("Tamanho de peso: %lu bytes\n", sizeof(paciente.peso));
    printf("Tamanho de altura: %lu bytes\n", sizeof(paciente.altura));
    printf("Tamanho de grau_miopia: %lu bytes\n", sizeof(paciente.grau_miopia));
    printf("Soma dos tamanhos das variáveis: %lu bytes\n", sizeof(paciente.idade) + sizeof(paciente.e_civil) + sizeof(
paciente.nome) + sizeof(paciente.peso) + sizeof(paciente.altura) + sizeof(paciente.grau_miopia));

    // o resultado da soma é 42, mas por conta do padding o tamanho de paciente.nome é 16
    // isso acontece porque uma variável do tipo char ocupa apenas 1 byte, então o vetor nome ocupa 10
    // para otimizar a quantidade de ciclos de clock, são adicionadas 6 posições de memória de modo ao vetor nome o
cupar 16 bytes
    // assim, o tamanho da struct dados_paciente é de 48 bytes

    printf("Tamanho da struct: %lu bytes\n", sizeof(struct dados_pacientes));

    return 0;
}

```