Nome: Filip Junio Felipe Prontuário: 1760319

Relatório das Aulas 1 e 2

```c
//Código da aula 1
//HelloWorld.c
#include <stdio.h>                          // Inclui a biblioteca stdio.h
int main() {                                // Método principal
  printf("Hello World!\n");                 // Msg
  return 0;                                 // Método int necessita de um retorno int
}
```

```c
//Código da aula 2
//estrutura_simples.c
#include <stdio.h>                                          //Importa a lib stdio.h
#define alturaMaxima 225                                    //Define a variável como 255

typedef struct                                              //Criação da estrtura
{
    int peso;   // peso em quilogramas                      //Declara a variável int peso na estrutura
    int altura; // altura em centimetros                    //Declara a variável int altura na estrutura

} PesoAltura;                                               //Nome da estrutura

int main()                                                  //Método Principal do programa
{
    int x;                                                  //Declara a variável int x
    PesoAltura pessoa1;                                     //Declara variável do tipo PesoAltura
    pessoa1.peso = 80;                                      //Atribui pessoa1.peso = 80
    pessoa1.altura = 185;                                   //Atribui pessoa1.altura = 185
    printf("Peso: %i, Altura %i. ", pessoa1.peso, pessoa1.altura);      //Mostra na tela os valores do Peso e da Altura
    if (pessoa1.altura > alturaMaxima){                     //Verifica se altura é maior que alturaMaxima(255)
        printf("Altura acima da maxima.\n");                //Se for maior, mostra na tela "Altura acima da maxima"
    }
    else{
        printf("Altura abaixo da maxima.\n");               //Se a altura não maior, mostra na tela "Altura abaixo da maxima."
    }
    printf("%p %p %p\n", &x, &pessoa1, &(pessoa1.altura));  //Mostra na tela os endereços de memória da variável X, Pessoa1 e Pessoa1.altura
    return 0;
}
```

```c
//Código da aula 2
//estrutura_simples2.c
#include <stdio.h>                                          //Importa a lib stdio.h
#include <malloc.h>                                         //Importa a lib malloc.h
#define alturaMaxima 225                                    //Define a variável como 255

typedef struct                                              //Criação da estrtura
{
    int peso;   // peso em quilogramas                      //Declara a variável int peso na estrutura
    int altura; // altura em centimetros                    //Declara a variável int altura na estrutura
} PesoAltura;                                               //Nome da estrutura

int main()                                                  //Método Principal do programa
{
    int x;                                                  //Declara a variável int x
    PesoAltura *pessoa1;                                    //Declara um ponteiro do tipo PesoAltura
    printf("Valor inicial do endereco: %p\n", pessoa1);     //Mostra na tela o endereço de pessoa1
    pessoa1 = (PesoAltura *)malloc(sizeof(PesoAltura));     //Aloca na memória o ponteiro
    printf("Peso: %i, Altura %i. ", pessoa1->peso, pessoa1->altura);    //Mostra na tela valor do peso e altura através do ponteiro
    pessoa1->peso = 80;                                     //Atribui peso=80 através do ponteiro
    pessoa1->altura = 185;                                  //Atribui altura = 185 através do ponteiro

    printf("Peso: %i, Altura %i. ", pessoa1->peso, pessoa1->altura);    //Mostra na tela valor do peso e altura através do ponteiro
    if (pessoa1->altura > alturaMaxima)                     //Verifica se altura é maior que alturaMaxima(255)
    {
        printf("Altura acima da maxima.\n");                //Se for maior, mostra na tela "Altura acima da maxima"
    }
    else{
        printf("Altura abaixo da maxima.\n");               //Se a altura não maior, mostra na tela "Altura abaixo da maxima."
    }

    printf("Enderecos: %p %p %p\n", &x, &pessoa1, pessoa1);     //Mostra na tela os endereços de memória
    return 0;
}
```

```c
//Código da aula 2
//testa_estrutura.c
#include <stdio.h>                          //Importa a lib stdio.h
#include <malloc.h>                         //Importa a lib malloc.h

typedef struct                              //Criação da estrutura
{
  int c1;                                   //Declara a variável int peso na estrutura
  int c2;                                   //Declara a variável int peso na estrutura
} TESTE, *PONT;                             //Nome da estrutura

PONT copiar(TESTE t1){                      //Método copiar receber uma estrutura tipo TESTE
    PONT x = (PONT) malloc(sizeof(TESTE));  //Declara a variável do tipo TESTE alocando na memória
    *x = t1;                                //Copia t1
    return x;                               //retorna um tipo PONT
}

int main(){                                 //Método Principal do programa
    TESTE y;                                //Declara variável do tipo TESTE
    y.c1 = 10;                              //Atribui c1 = 10
    y.c2 = 22;                              //Atribui c2 = 22
    PONT w = copiar(y);                     //Declara variável w recebendo os dados através do metódo copiar

    printf("c1: %i, c2: %i\n", w->c1, w->c2);       //Mostra na tela os valores de c1 e c2
    return 0;
}
```