

LISTA 12 – 21/06

QUESTÃO 1

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

int main(){
    int motor[12][2],cl[12][2], l, c,lucro,i;
    for (c=0; c<2; c++){
        for (l=0; l<12; l++){
            printf ("Numero do motor %d no mes %d: ", c+1, l+1);
            scanf ("%d", &motor[l][c]);
        }
    }
    printf ("\n");
    for (l=0; l<2; l++){printf ("Custo do motor %d: ", l+1);scanf ("%d",
&cl[l][0]);}
    printf ("\n");
    for (l=0; l<2; l++){printf ("Lucro do motor %d: ", l+1);scanf ("%d",
&cl[l][1]);}
    FILE * controle = fopen ("Controle.txt", "w");
    for (i=0; i<12; i++){
        fprintf(controle,"\nMes (%d) ", (i+1) );
        fprintf(controle,"\nCusto motor 1: %d ", (motor[i][0]*cl[0][0]) );
        fprintf(controle,"\nLucro motor 1: %d ", (motor[i][0]*cl[0][1]) );
        fprintf(controle,"\nCusto motor 2: %d ", (motor[i][1]*cl[1][0]) );
        fprintf(controle,"\nLucro motor 2: %d \n ", (motor[i][1]*cl[1][1]) );
        lucro+= ((motor[i][0]*cl[0][1]) + (motor[i][1]*cl[1][1]));
    }
    printf ( "\nLucro total: %d",lucro);
    fclose (controle);
    return 0;
}
```

QUESTÃO 2

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

int main(){
    int cabelo[5][3],unha[5][3],pagamento[6]={0,0,0,0,0,0},l,c;
    char manicure[5][34];
    for (l=0;l<5;l++){
        printf("Nome da manicure %i: ",l+1);
```

```

scanf("%s",manicure[l]);
}
for (c=0; c<3; c++){
    for (l=0; l<5; l++){
        if(c==0){printf ("Pedicure pela manicure %d: ", l+1);
        scanf ("%d",&cabelo[l][c]);}
        else if(c==1){printf ("Manicure pela manicure %d: ",l+1);
        scanf ("%d", &cabelo[l][c]);}
        else{printf ("Podologia pela manicure %d: ", l+1);
        scanf ("%d", &cabelo[l][c]);}
    }
}
printf("\e[H\e[2J");
for (l=0; l<5; l++){
    for (c=0; c<3; c++){
        if (c==0){unha[l][c]=cabelo[l][c]*15;}
        else if(c==1){unha[l][c]=cabelo[l][c]*20;}
        else{
            unha[l][c]=cabelo[l][c]*30;
            pagamento[l]+=unha[l][c];}
    }
}
for (l=0; l<5; l++){
    printf ("\n");
    for (c=0;c<3;c++){
        printf ("%d |",unha[l][c]);
    }
}
printf ("\n\n");
for (l=0; l<5; l++){printf ("%d |",pagamento[l]);}
FILE*arq=fopen("Faturamento.txt", "w");
if (arq == NULL){
    printf("ERRO..\n");
}else{
    printf("\n0 arquivo aberto!");}
for (l=0; l<5; l++){
    fprintf(arq, "\n%s R$d", manicure[l],pagamento[l]);
}
fclose (arq);
return 0;
}

```

QUESTÃO 3

Valores são aqueles em que é definido um parâmetro para determinada variável, podendo ser aplicados métodos e armazenar valores. Já os registros, são registros/estruturas que armazenam

informações de vários tipos, sendo dados desiguais agrupados sem conter métodos. O typedef tem a função de criar nomes para dados sem criar novos tipos, e ele é buscado através do nome da estrutura:

```
typedef struct Habitante{
    char sexo, olhos;
    float altura;
    int idade;
}Habitante;
```

QUESTÃO 4

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

```
typedef struct Habitante{
    char sexo, olhos;
    float altura;
    int idade;
}Habitante;

void Pesquisa (Habitante *habitantes){
    printf ("Idade: ");
    scanf ("%i", &habitantes->idade);
    printf ("Altura: ");
    scanf ("%f", &habitantes->altura);
    printf ("Sexo - (F) ou (M): ");
    scanf (" %c", &habitantes->sexo);
    printf ("Cor do olho - azul(a), verde(v) ou castanho(c): ");
    scanf (" %c", &habitantes->olhos);
}

void TesteCastanho (Habitante habitantes[]){
    float mediaalt=0, somaalt=0, num;
    for (int i=0; i<50; i++){
        if (habitantes[i].olhos=='c' && habitantes[i].altura>160){
            somaalt+=habitantes[i].idade;
            num++;
        }
    }
    if(somaalt>0){mediaalt=somaalt/num;}
    printf ("\nMédia das idades de pessoas com olhos castanhos e acima de 160cm: %.2f", mediaalt);
}

void MaiorId (Habitante habitantes[]){
    int maisidade=0;
    for (int i=0; i<50; i++){
```

```

        if (habitantes[i].idade>maisidade){maisidade=habitantes[i].idade;}
    }
    printf ("\nMaior idade entre os habitantes: %i", maisidade);
}
void testefem (Habitante habitantes[]){
    int cond_fem=0;
    for (int i=0; i<50; i++){
        if (habitantes[i].sexo=='f' && habitantes[i].idade>=20 &&
habitantes[i].idade<=45){cond_fem++;}
        else if (habitantes[i].sexo=='f' && habitantes[i].olhos=='v' &&
habitantes[i].altura<170){cond_fem++;}
        else{cond_fem=0;}
    }
    printf ("\nA quantidade de mulheres entre 20 e 45 anos ou com olhos
verdes e altura menor que 170cm é: %i", cond_fem++);
}
void PercentualHomens(Habitante habitantes[]){
    float percent=0, homens=0;
    for (int i=0; i<50; i++){if (habitantes[i].sexo=='m'){homens++;}}
    percent=(homens*100)/50;
    printf ("\nPercentual de homens: %.1f%%", percent);
}
int main(void) {
    Habitante habitantes[50];
    int maisidade;
    for (int i=0; i<50; i++){printf ("\n%d. ",
i+1);Pesquisa(&habitantes[i]);}
    TesteCastanho(habitantes);
    MaiorId(habitantes);
    PercentualHomens(habitantes);
    testefem(habitantes);
    return 0;
}

```

QUESTÃO 5

```

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

typedef struct Chapa{
    char nome[34], responsavel[30];
    int votos;
}Chapa;
void Cadastro(Chapa *chapas){
    printf("Nome da chapa: ");
    gets(chapas->nome);
}

```

```

    printf ("Responsavel: ");
    gets(chapas->responsavel);
}
void Votosnum(Chapa *chapas){
    int voto[100];
    chapas[0].votos=0, chapas[1].votos=0;
    for (int i=0; i<4; i++){
        printf ("Seu voto: (1) para a equipe [%s] / (2) para a equipe [%s]:", chapas[0].nome, chapas[1].nome);
        scanf ("%i", &voto[i]);
        switch (voto[i]){
            case 1:
                chapas[0].votos++;
                break;

            case 2:
                chapas[1].votos++;
                break;

            default:
                printf ("\nerro...");
                break;
        }
    }
    if (chapas[0].votos>chapas[1].votos){
        printf ("\nCampeã: Chapa-%s / Responsavel-%s / Votos-%i ", chapas[0].nome, chapas[0].responsavel, chapas[0].votos);}
    else if (chapas[0].votos<chapas[1].votos){
        printf ("\nCampeã: Chapa-%s / Responsavel-%s / Votos-%i ", chapas[1].nome, chapas[1].responsavel, chapas[1].votos);}
    else {
        printf ("\nChapas empatadas: %i votos / %i votos totais", chapas[1].votos, chapas[0].votos+chapas[1].votos);}
}
int main(void) {
    Chapa chapas[2];
    for (int i=0; i<2; i++){
        printf ("\n%i. ", i+1);
        Cadastro(&chapas[i]);
    }
    Votosnum(chapas);
    return 0;
}

```

QUESTÃO 6

```

#include <stdio.h>
#include <stdlib.h>

```

```

#include <string.h>

typedef struct Paciente{
    int codigo, telefone;
    char nome[30], end[30];
}Paciente;
typedef struct medicos{
    int codigo, telefone;
    char nome[30], end[30];
}medicos;
typedef struct consultas{
    int codigo;
}consultas;
void Cadastro (Paciente *pacientes, int i){
    printf ("Nome: ");
    gets (&pacientes->nome);
    if (i>0){gets (&pacientes->nome);}
    printf ("Endereço: ");
    gets (&pacientes->end);
    printf ("Telefone: ");
    scanf ("%i", &pacientes->telefone);
    pacientes->codigo=i;
}
void Pacientes (Paciente pacientes[], int i){
FILE * arq = fopen ("cliente.txt", "a+");
    if (arq == NULL){printf("ERRO...\n");}
    else{printf("\nO arquivo foi aberto!");}
    fprintf(arq, "\nCodigo: %i \nNome: %s \nEndereço: %s \nTelefone: %i",
pacientes[i].codigo, pacientes[i].nome, pacientes[i].end,
pacientes[i].telefone);
    fclose(arq);
}
int main(void) {
    Paciente pacientes[10];
    for (int i=0; i<3; i++){
        printf ("\n%d. ", i+1);
        Cadastro(&pacientes[i], i);
        Pacientes(pacientes, i);
    }
    return 0;
}

```

QUESTÃO 7

```

typedef struct Empregado{
    int matricula;
    char nome[100];
}

```

```

    float salario;
}
Empregado;
int numeroEmpregados = 0;
void lerEmpregadosArquivo(Empregado VetorEmpregados[]);
void ListaCompleta(Empregado VetorEmpregados[]);
void strSplit(char *strTosplit, char *strArr[], char *strSeparet, int
nArr);
int main(){
    Empregado VetorEmpregados[1000];
    lerEmpregadosArquivo(VetorEmpregados);
    ListaCompleta(VetorEmpregados);
    printf("Fim do programa!\n");
    return 0;
}
void strSplit(char *strTosplit, char *strArr[], char *strSeparet, int
nArr){
    int i = 0;
    char * pch;
    pch = strtok (strTosplit, strSeparet);
    for(i = 0; i < nArr; i++){
        strArr[i] = pch;
        pch = strtok (NULL, strSeparet);
    }
}
void lerEmpregadosArquivo(Empregado VetorEmpregados[]){
    FILE *arquivo;
    arquivo = fopen("empregados.txt", "r");
    char linha[100];
    char *result;
    char * informacoes_linha[3];
    int i = 0;
    while (!feof(arquivo)) {
        result = fgets (linha, 100, arquivo);
        strSplit(linha, informacoes_linha, ";", 3);
        VetorEmpregados[i].matricula = atoi(informacoes_linha[0]);
        strcpy(VetorEmpregados[i].nome, (const char*)(informacoes_linha[1])
);
        VetorEmpregados[i].salario = atof(informacoes_linha[2]);
        i++;
    }
    numeroEmpregados = i;
}
void ListaCompleta(Empregado VetorEmpregados[]){
    int i;
    for (i = 0; i < numeroEmpregados; i++){
        printf(" %s : %d : %.2f \n", VetorEmpregados[i].nome,
VetorEmpregados[i].matricula, VetorEmpregados[i].salario);
    }
}

```

