

Lista 02

Jefferson Avilar

1 - Construa um algoritmo que apresente o nome e o salário de dois funcionários, de acordo com os seguintes critérios:

- a) Salários que sejam maiores ou iguais a R\$ 1000,00 e menores ou iguais a R\$ 1500,00
- b) Funcionários pertencentes aos departamentos de produção ou engenharia.

Obs: Os departamentos são reconhecidos pelas letras (P) Produção e (E) Engenharia

São fornecidos o nome do funcionário (NF), o seu salário (SAL) e o departamento onde trabalha (DEP).

```
#include <stdio.h>

int main()
{
    char nf1, nf2, dep1, dep2;
    int sal1, sal2;

    printf("Insira o seu nome: "); scanf("%c", &nf1);
    printf("Insira seu salário: "); scanf("%d", &sal1);
    printf("E o seu departamento: "); scanf("%c", &dep1);

    return 0;
}
```

2 - Construa um algoritmo que calcule o novo salário (SAL_NOVO) de um funcionário. Considere que o funcionário deverá receber um reajuste de 15% caso seu salário (SAL) seja menor que 500. Se o salário for maior ou igual a 500, mas menor ou igual a 1000, o reajuste deve ser de 10%. Caso o salário seja maior que 1000, o reajuste deve ser de 5%.

```
#include <stdio.h>

int main()
{
    int sal;
    int novo_sal;

    scanf("%d", &sal);

    if (sal < 500)
    {
        novo_sal = sal+(sal*0.15);
        printf("Seu novo salário é %d\n", novo_sal);
    }

    if (sal >= 500 && sal <= 1000)
    {
```

```

    novo_sal = sal+(sal*0.10);
    printf("Seu novo salário é %d\n", novo_sal);
}

if (sal > 1000)
{
    novo_sal = sal+(sal*0.05);
    printf("Seu novo salário é %d\n", novo_sal);
}

return 0;
}

```

3 - Construa um algoritmo que leia dois números (A e B). Caso A seja igual a B, apresentar a soma dos dois. Caso um seja maior que o outro, apresentar a diferença entre os dois números (sempre lembrando que a diferença entre dois números é SEMPRE positiva)

```

#include <stdio.h>

int main(int argc, char const *argv[]) {

    int a, b;
    unsigned int z;

    scanf ("%d %d", &a, &b);

    z = a - b;

    if (a==b) printf("%d\n",a+b);

    else if (a < b) printf("%d\n", z*-1);
    else if (a > b) printf("%d\n", z);

    return 0;
}

```

4 - Construa um algoritmo que leia o código de um livro (CL) e apresente a categoria do livro, conforme a tabela abaixo:

Código do Livro (CL)	Categoria
A	Ficção
B	Não-Ficção
Qualquer outro código	Invalido

```

#include <stdio.h>

```

```

int main()
{
    char cl;

    scanf("%c", &cl);

    if (cl == 'a' || cl == 'A') printf("Ficção\n");
    else if (cl == 'b' || cl == 'B') printf("Não-Ficção\n");
    else printf("Inválido\n");

    return 0;
}

```

5 - Construa um algoritmo que leia a quantidade de dinheiro existente no caixa de uma empresa (CAIXA), a quantidade de produtos a ser comprada (QTD) e o preço de cada unidade (PR). Caso o valor total da compra seja superior a 80% do valor em caixa, a compra deve ser feita a prazo (3x), com juros de 10% sobre o valor total. Caso contrário, a compra deverá ser realizada a vista, onde a empresa receberá 5% de desconto. Apresentar a forma de pagamento escolhida e o valor a ser pago (total a vista ou total a prazo), dependendo da escolha realizada pelo programa.

```

#include <stdio.h>

int main()
{
    float caixa, pr;
    int qtd = 0;
    float valor_total;

    printf("Insira o valor em caixa: "); scanf("%f", &caixa);
    printf("Insira o quantidade de produtos: "); scanf("%d", &qtd);
    printf("O preço da unidade: "); scanf("%f", &pr);

    valor_total = qtd * pr;

    if (valor_total > (caixa * 0.8))
    {
        printf("A compra será feita em 3x, com juros de 10(por cento), e será no valor de R$ %.2f reais\n", (valor_total + valor_total * 0.1));
        printf("A parcela será de R$ %.2f\n", (valor_total + valor_total * 0.1)/3);
    }
    else
    {
        printf("A compra será feita a vista, com desconto de 5(por cento), valor = %.2f\n", (valor_total - valor_total * 0.05));
    }

    return 0;
}

```

6 - Construa um algoritmo que leia as informações de: horas trabalhadas (HT), valor da hora trabalhada (VH). Calcule e apresente o salário líquido do empregado, baseado nas tabelas abaixo.

OBS: Salário Líquido = Salário Bruto – INSS – Imposto de Renda

Salário Bruto = Horas trabalhadas * Valor da hora trabalhada

INSS = 11% do salário bruto

Imposto de Renda

após descontar o INSS usar esse valor e ler a alíquota do imposto de renda e parcela a deduzir na tabela abaixo

Salário Bruto - INSS	Alíquota	Valor a deduzir
Até \$1.257,12	Isento: 0%	
De \$1.257,13 até \$2.512,08	15%	\$188,57
Mais que \$2.512,08	27,5%	\$502,58

OBS: Imposto de Renda = Alíquota * (Salário Bruto – INSS) – Valor a Deduzir

```
#include <stdio.h>

int main()
{
    int horas_trabalhadas;
    long double aliquota, valor_hora, salario_bruto, salario_liquido, inss,
    imposto_de_renda;

    printf("Insira quantas horas foram trabalhadas: "); scanf("%d",
    &horas_trabalhadas);
    printf("E quanto vale a hora trabalhada: "); scanf("%Lf", &valor_hora);

    salario_bruto = horas_trabalhadas * valor_hora;
    inss = salario_bruto * (0.11);

    if (salario_bruto <= 1257.1)
    {
        salario_bruto = salario_bruto - inss;
    }

    if (salario_bruto >= 1257.13 && salario_bruto <=2512.08)
    {
        aliquota = salario_bruto * 0.15;
        imposto_de_renda = aliquota * (salario_bruto - inss) - (188.57);
    }
    if (salario_bruto > 2512.08)
    {
        aliquota = salario_bruto * (0.275);
        imposto_de_renda = aliquota * (salario_bruto - inss) - (502.58);
    }

    salario_liquido = salario_bruto - inss - imposto_de_renda;

    printf("Seu salário real(liquido) é R$ %.2Lf\n", salario_liquido);

    return 0;
}
```

7 - Repita o exercício 14, só que agora, a porcentagem de desconto de INSS não é mais fixa. O

desconto acontece de acordo com a tabela abaixo:

Salário Bruto	Alíquota
Até \$800,45	7,65%
De \$800,46 até \$900,00	8,65%
De \$900,01 até \$1.334,07	9,00%
De \$1.334,08 até \$2.668,15	11,00%

8 - Construa um algoritmo que calcule e apresente quanto deve ser pago por um produto considerando a leitura do preço de etiqueta (PE) e o código da condição de pagamento (CP). Utilize para os cálculos a tabela de condições de pagamento a seguir:

Código da condição de pagamento	Condição de pagamento
1	À vista em dinheiro ou cheque, com 10% de desconto
2	À vista com cartão de crédito, com 5% de desconto
3	Em 2 vezes, preço normal de etiqueta sem juros
4	Em 3 vezes, preço de etiqueta com acréscimo de 10%

```
#include <stdio.h>

int main()
{
    float pe;
    int cp;

    printf("Insira o preço de etiqueta: "); scanf("%f", &pe);
    printf("Insira a condição de pagamento [1-4]: "); scanf("%d", &cp);

    if (cp == 1) printf("À vista com 10% de desconto = R$ %.2f\n", pe-(pe*0.1));
    if (cp == 2) printf("À vista com cartão de crédito = R$ %.2f\n", pe-(pe*0.05));
    if (cp == 3) printf("Em 2 vezes, parcelas de R$ %.2f\n", pe/2);
    if (cp == 4) printf("Em 3 vezes, parcelas de R$ %.2f\n", pe/3);

    return 0;
}
```

9 - Construa um algoritmo que tendo como dados de entrada o preço de um produto (PR) e seu código de origem (CO), apresente o preço e a sua procedência, de acordo com a tabela abaixo:

Código de Origem (CO)	Procedência
1	Sul
2	Sudeste
3	Centro-Oeste
4	Norte
5	Nordeste

```
#include <stdio.h>

int main()
{

    int pr, co;

    printf("Insira o preço do produto: "); scanf("%d", &pr);
    printf("Insira o código de origem do produto [1-5]: "); scanf("%d", &co);

    if (co == 1) printf("Sul\n");
    if (co == 2) printf("Sudeste\n");
    if (co == 3) printf("Centro-Oeste\n");
    if (co == 4) printf("Norte\n");
    if (co == 5) printf("Nordeste\n");

    return 0;
}
```

10 - Construa um algoritmo que, dados os comprimentos dos três lados (A, B e C) de um triângulo, verifique o tipo de triângulo formado. Apresentar qual é o tipo. Sabe-se que:

- a) Triângulo do tipo Equilátero – possui os três lados iguais
- b) Triângulo do tipo Isósceles – possui dois lados iguais
- c) Triângulo do tipo Escaleno – possui os três lados diferentes

```
#include <stdio.h>

int main()
{

    int a, b, c;

    printf("Insira os três valores valores separados por um espaço: "); scanf("%d %d %d", &a, &b, &c);

    if (a != b && a != c && b != c) printf("Escaleno\n");
```

```

else if (a == b && a == c && b == c) printf("Equilátero\n");
else printf("Isósceles\n");

return 0;
}
}

```

11 - Construa um algoritmo que, dada a idade de um nadador (ID), classifique-o em uma das seguintes categorias e apresente a categoria:

Idade (ID)	Categoria
5 até 7 anos	Infantil A
8 até 10 anos	Infantil B
11 até 13 anos	Juvenil A
14 até 17 anos	Juvenil B
Acima de 18 anos	Adulto

```

#include <stdio.h>

int main()
{
    int id;

    printf("Insira sua idade: "); scanf("%d", &id);

    if (id >= 5 && id <= 7) printf("Categoria --> Infantil A\n");
    if (id >= 8 && id <= 10) printf("Categoria --> Infantil B\n");
    if (id >= 11 && id <= 13) printf("Categoria --> Juvenil A\n");
    if (id >= 14 && id <= 17) printf("Categoria --> Juvenil B\n");
    if (id >= 18) printf("Categoria --> Adulto");

    return 0;
}

```

12 - Faça um algoritmo que leia 2 valores numéricos e um símbolo. Caso o símbolo seja um dos relacionados abaixo efetue a operação correspondente com os valores. Atenção para a divisão por 0!

- “+” → operação de soma;
- “-” → operação de subtração;
- “*” → operação de multiplicação;
- “/” → operação de divisão;

```
#include <stdio.h>

int main()
{
    char zx;
    int n1, n2;

    printf("***** Insira qual operação deseja fazer *****\n");
    printf("'+' para Adição\n'-' para Subtração\n'*' para Multiplicação\n'/' para Divisão\n");
    scanf("%c", &zx);

    printf("Insira o primeiro valor: "); scanf("%d", &n1);
    printf("Insira o segundo valor: "); scanf("%d", &n2);

    if (zx == '+') printf("Valor da soma entre %d e %d = %d\n", n1, n2, (n1+n2));
    else if (zx == '-') printf("Valor da subtração entre %d e %d = %d\n", n1, n2, (n1-n2));
    else if (zx == '*') printf("Valor da multiplicação entre %d e %d = %d\n", n1, n2, (n1*n2));
    else if (zx == '/') printf("Valor da divisão entre %d e %d = %d\n", n1, n2, (n1/n2));

    return 0;
}
```
