

Aula Prática 04 (Revisão condicional, repetição, uso função)

• **Instruções:**

- Os exercícios deverão ser feitos em aula de laboratório durante o tempo da aula;
- O professor irá esclarecer dúvidas em aula;
- Crie uma pasta com seu nome e vá gravando seus programas implementados.

Exercício 1

Uma empreiteira paga seus pedreiros por metros quadrados de serviços produzidos **diariamente**, adicionando também o valor do almoço e passagem para irem e voltar do trabalho. Essa construtora procurou você para desenvolver um sistema que calcula quanto ela deve pagar para cada um de seus pedreiros. Apresente um programa que resolva o problema citado.

- O programa deverá receber:

- O valor do vale-alimentação;
- O valor do transporte unitário (sendo que cada dia o pedreiro usa 2 vales);
- Quantos metros quadrados trabalhou;

- O valor do metro quadrado é calculado da seguinte forma:

Quantos metros quadrados trabalhou?	Valor do metro quadrado
Menos de 10m2	R\$ 10,00
Mais ou igual a 10m2 até 20m2	R\$ 11,50
Mais ou igual a 20m2	R\$ 13,00

- Crie a lógica para calcular o valor a ser pago ao pedreiro;
- Depois na função principal, receba os valores especificados, execute os cálculos e armazene o valor do pagamento;
- Ao final, o programa deve imprimir a seguinte mensagem ao final de sua execução: "O pedreiro deve receber R\$ x.", onde x é o valor a ser pago.

Execute o código para testar seu programa de computador e verificar se o resultado está correto. [salve o seu código com o nome: **ap4ex1.c**]

Extra: ao final faça uma estrutura de repetição, onde usuário informa quantos pedreiros quer processar os dados e faz o mesmo processamento para N vezes (quantos forem os pedreiros).

Exercício 2

Esta questão avaliará os conhecimentos do aluno quanto a utilização das estruturas de repetição (laços) e de decisão. Elabore um programa que deve encontrar os números ímpares e deverá imprimir todos os números ímpares entre 0 e um valor máximo (numeroMax), a ser informado como entrada pelo usuário.

- a) Você deverá elaborar uma FUNÇÃO que deverá verificar se o número é ímpar (a função deverá receber o valor corrente entre 0 e numeroMax e retornar 0 se não for ímpar e 1 se for ímpar).
- b) Ao final mude seu programa de forma que ele deve imprimir apenas 10 números por linha.

Execute o código para testar seu programa de computador e verificar se o resultado está correto. [salve o seu código com o nome: **ap4ex2.c**]

Exercício 3

Agora você deverá desenvolver um programa que irá verificar se o aluno foi aprovado ou não nas disciplinas que cursou.

O que o software deverá fazer:

- a) Receber o número de disciplinas que o aluno cursou.
- b) Para cada disciplina deverá ser feito o seguinte:
 - Receber o número de aulas lecionadas (por exemplo, 60 aulas);
 - Receber o número de presenças que o aluno teve (por exemplo, 55);
 - Receber a nota que o aluno tirou na disciplina (entre 0 e 100);
 - Verificar se o aluno foi reprovado por frequência (regra: frequência deve ser maior ou igual a 75% das aulas lecionadas) – Faça a função `aprovadoFrequencia`, que recebe a quantidade de aulas e a quantidade de presença e retorna se foi reprovado por frequência (valor 0) ou aprovado (valor 1);
 - Caso o aluno não tenha sido reprovado por frequência, verificar se ele foi aprovado (após já ter sido aplicado o exame especial), com a seguinte regra:
 - Nota < 60: reprovado;
 - Nota >= 60: aprovado.
- c) Faça uma função que verifique o conceito do aluno na disciplina dada a nota que ele tirou (ou seja, a função `conceito` recebe a nota e converte para um conceito A até F)
- d) Ao final do seu programa imprima:
 - a. O aluno foi [aprovado / reprovado] na disciplina X [1, 2...] com conceito [A, B, ... F]; ou
 - b. O aluno foi reprovado por frequência, seu conceito é I (infrequente).

Sugestões:

- Pode fazer seu código primeiro para uma disciplina apenas, depois de testado pode incluir a estrutura que vai receber mais disciplinas (no início quando usuário informar e alterar o código para contemplar várias disciplinas).

Segue tabela referência de conceitos:

Pontuação	Conceito
90 a 100	A - Excelente
80 a 89	B - Ótimo
70 a 79	C - Bom
60 a 69	D - Regular
40 a 59	E - Fraco
0 a 39 ou infrequencia	F - Insuficiente

Execute o código para testar seu programa de computador e verificar se o resultado está correto. [salve o seu código com o nome: **ap4ex3.c**]

Exercício 4 – Passagem de Parâmetros

Para reforçar o conceito de ponteiro e passagem de parâmetros por valor e por referência, faça o seguinte: crie 2 procedimentos (void somandoValor e void somandoReferencia), que deverão somar 2 números e adicionar ao valor corrente de uma variável somaValor (somaValor = somaValor + a + b) e mesma coisa para somaReferência, que usará o conceito de ponteiro.

O código MAIN será o seguinte:

```
int main(int argc, char *argv[]) {
    int x = 10;
    int y = 5;
    int somaValor = 0;
    int somaReferencia = 0;
    somandoValor(x,y, somaValor);
    somandoReferencia(x,y, &somaReferencia);
    printf("MAIN - Valor de somaValor = %d\n",somaValor);
    printf("MAIN - Valor de somaReferencia = %d\n",
somaReferencia);
    return 0;
}
```

Os procedimentos deverão ter as seguintes assinaturas:

void somandoValor (int a, int b, int somaValor)

void somandoReferencia (int a, int b, int *somaReferencia)

- Imprima o valor de somaValor dentro do procedimento e de somaReferencia também e compare com os valores impressos no MAIN. Isso comprovará o que vimos dos conceitos de passagem de parâmetros e ponteiro.

Execute o código para testar seu programa de computador e verificar se o resultado está correto. [salve o seu código com o nome: **ap04-ex4.c**]