



Universidade Federal do Rio Grande do Norte – UFRN  
Escola Agrícola de Jundiaí – EAJ  
Curso Técnico em Informática  
Programação de Computadores  
Profa. Alessandra Mendes

**LISTA 9 – 06/04/2021**

Conteúdo: Funções

**Elabore os seguintes algoritmos:**

1. Numa disciplina são ministradas duas provas e dois trabalhos, mas a média é calculada considerando-se apenas a maior nota de prova e a maior nota de trabalho. Elabore um algoritmo que, dadas as quatro notas de um aluno, informe a sua média. Para isso, utilize duas funções: uma que determine o maior valor entre as notas (de provas ou de trabalhos) e outra que calcule a média aritmética das duas maiores notas.
2. Faça uma função que receba um número inteiro por parâmetro e retorne verdadeiro se ele for par ou falso se for ímpar.
3. Escreva uma função que receba 2 números inteiros  $n1$  e  $n2$  como entrada e retorne a soma de todos os números inteiros contidos no intervalo  $[n1, n2]$ . Use esta função em um programa que leia  $n1$  e  $n2$  do usuário e imprima a soma.
4. Escreva um programa que leia um valor inteiro (maior do que 1 e menor ou igual a 10) e exiba a tabuada (até 10) de multiplicação do número lido. Você deverá escrever as seguintes funções:  
- `int LeNumero(int n1, int n2)`  
Lê um número inteiro no intervalo especificado ( $n1$ ,  $n2$ ) e o retorna. Cada vez que for digitado um número inválido (fora do intervalo especificado) a função deve exibir a mensagem "Número inválido. Digite novamente!"  
- `void Tabuada(int n)`  
Recebe como parâmetro um número inteiro e exibe na tela a tabuada de multiplicação até 10 do número lido. Exemplo: número lido 5  
5 x 1 = 5      5 x 2 = 10      ....      5 x 10 = 50
5. Elabore um algoritmo que leia o raio de um círculo. Em seguida, elabore duas funções diferentes, uma chamada *area* que calcula e retorna a área do círculo e a outra chamada *perimetro* que calcula e retorna o perímetro do círculo.  
Considere: Área =  $PI * r^2$ ; Perímetro =  $PI * 2 * r$ .
6. Elaborar um algoritmo que leia um vetor de 10 posições de inteiros e chame uma função MinMax. Esta função deve receber o vetor lido e, por referência, duas variáveis inteiras, min e max. O objetivo da função é buscar os valores do menor e maior elementos do vetor e atribuir às variáveis min e max respectivamente, retornando-as e escrevendo na tela após o retorno da função.
7. Escreva uma função chamada Troca que troque os valores dos parâmetros recebidos. Sua assinatura deve ser: `void troca(float *a, float *b)`.

8. Escreva uma função que determine a média e a situação de um aluno em uma disciplina. A função recebe como parâmetros as três notas de um aluno (p1, p2, e p3), seu número de faltas (faltas), o número total de aulas da disciplina (aulas) e o ponteiro para uma variável (media), conforme a seguinte assinatura:

`char situacao(float p1, float p2, float p3, int faltas, int aulas, float *media);`

Na variável indicada pelo ponteiro media, a função deve armazenar a média do aluno, calculada como a média aritmética das três provas. Além disso, a função deve retornar um caractere indicando a situação do aluno no curso, definido de acordo com o seguinte critério:

Número de Faltas	Média	Situação	Retorno
Menor ou igual a 25% do total de aulas	Maior ou igual a 6,0	Aprovado	<b>A</b>
	Menor que 6,0	Reprovado	<b>R</b>
Maior que 25% do total de aulas	Qualquer	Reprovado por faltas	<b>F</b>

Em seguida, escreva a função principal de um programa que utiliza a função anterior para determinar a situação de um aluno. O programa deve:

- Ler do teclado três números reais e dois números inteiros, representando as notas da p1, p2 e p3, o número de faltas e o número de aulas, respectivamente;
- Chamar a função desenvolvida na primeira questão para determinar a média e a situação do aluno na disciplina;
- Exibir a média (com apenas uma casa decimal) e a situação do aluno, isto é, "APROVADO", "REPROVADO" ou "REPROVADO POR FALTAS", dependendo do caractere retornado pela função, conforme a tabela acima.

9. Crie um algoritmo para manipular vetores. O seu programa deve implementar uma função chamada `inverteVetor`, que recebe como parâmetro dois vetores V1 e V2, ambos de tamanho N. A função deve copiar os elementos de V1 para V2 na ordem inversa. Ou seja, se a função receber  $V1 = \{1,2,3,4,5\}$ , a função deve copiar os elementos para V2 na seguinte ordem:  $V2 = \{5,4,3,2,1\}$ . Além disso, a função também deve retornar o maior valor encontrado em V1. A função deve possuir a seguinte assinatura:

`int inverteVetor(int *v1, int *v2, int n);`

Elabore um teste da função no main.

10. Em seguida, utilizando o algoritmo anterior implemente outra função chamada `multiplicaEscalar`, que recebe como parâmetro dois vetores V1 e V2 (ambos de tamanho N), e um número inteiro X. A função deve multiplicar cada um dos elementos de V1 por X e armazenar os resultados em V2. A função deve possuir a seguinte assinatura:

`void multiplica_escalar(int *v1, int *v2, int x, int n);`

No main, utilize as funções `inverteVetor` e `multiplicaEscalar` para inverter um vetor de tamanho 10 fornecido pelo usuário e em seguida multiplicar esse vetor por um escalar também fornecido pelo usuário. Por último, o programa deverá exibir o vetor resultante.