

# TECNÓLOGO EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

## PROGRAMAÇÃO DE COMPUTADORES II

*Revisão do Conteúdo: Programação de Computadores I*  
*Avaliação de Conhecimento*



|                   |                                                        |
|-------------------|--------------------------------------------------------|
| <b>Ruim:</b>      | <b>resolução das questões 1 e 2;</b>                   |
| <b>Regular:</b>   | <b>resolução das questões 1, 2, 3 e 4;</b>             |
| <b>Bom:</b>       | <b>resolução das questões 1, 2, 3, 4, 5 e 6;</b>       |
| <b>Muito Bom:</b> | <b>resolução das questões 1, 2, 3, 4, 5, 6, 7 e 8;</b> |
| <b>Excelente:</b> | <b>resolução de todas as questões;</b>                 |

- 
1. Faça um programa em C que solicite ao usuário um valor inteiro, eleve-o ao quadrado e apresente o resultado. Após isso solicite ao usuário, novamente, um segundo valor inteiro, eleve-o ao cubo e apresente o resultado. Por fim solicite ao usuário um terceiro valor inteiro, com esse valor, some o quadrado calculado anteriormente com o cubo, também calculado, e divida esta soma pelo terceiro valor, mostrando o resultado final da operação. Veja o exemplo:

- Digite o primeiro valor: 5  
- O valor ao quadrado = 25

- Digite o segundo valor: 10  
- O valor ao cubo = 1000

- Digite o terceiro valor: 5  
- Cálculo final = 205

2. Faça um programa em C que solicite apenas “um” valor inteiro ao usuário contendo três dígitos (considere que o usuário sempre digitará um valor correto, contendo três dígitos). Agora crie um código que efetue a soma desses três dígitos (esse é para pensar). Veja o exemplo:

- Digite um valor inteiro com 3 dígitos: 937  
- Soma dos seus dígitos =  $9 + 3 + 7 = 19$

3. Faça um programa em C que indique a ordem de classificação para quatro atletas olímpicos de acordo com o seu tempo de chegada na prova dos 100m de natação. Considerando que a ordem de classificação dos nadadores é feita na ordem crescente dos seus tempos, ou seja, o primeiro é quem possui o menor tempo, solicite ao usuário os quatro tempos (*double*) e apresente a classificação final. Veja o exemplo:

- Digite os quatro tempos: 2.35 2.20 3.05 2.90  
- Classificação Final:

- 1º lugar – “2.20”
- 2º lugar – “2.35”
- 3º lugar – “2.90”
- 4º lugar – “3.05”

(substitua os valores ente “” pelos digitados pelo usuário)

4. Faça um programa em C que solicite ao usuário um valor inteiro, após isso calcule o fatorial desse valor. Utilize uma variável do tipo Double para armazenar o resultado do fatorial, a saída do seu programa deve obedecer ao padrão apresentado a seguir. Veja o exemplo:

- Digite um valor: 4
- Fatorial(“4”) = “4 x 3 x 2 x 1 = 24”
  
- Digite um valor: 6
- Fatorial(“6”) = “6 x 5 x 4 x 3 x 2 x 1 = 720”

(substitua os valores ente “” pelos digitados pelo usuário e descobertos por você)

5. Faça um programa em C que receba repetidamente um número inteiro, até que o seu valor seja “-1” (indica sair). A cada número digitado pelo usuário o programa deve identificar e apresentar se trata-se de um número primo ou não. Veja o exemplo:

- Digite o valor inteiro: 5
- “5” é PRIMO!
  
- Digite o valor inteiro: 4
- “4” NÃO é PRIMO!
  
- Digite o valor inteiro: 13
- “13” é PRIMO!
  
- Digite o valor inteiro: -1

(substitua os valores ente “” pelos digitados pelo usuário e descobertos por você)  
(primos são números divisíveis apenas por um “1” e por ele mesmo)

6. Faça um programa em C que gere aleatoriamente um vetor contendo 10 elementos inteiros com valores variando entre 0 e 9. Após isso exiba o vetor aleatório gerado e ordene-o de maneira crescente. Exiba também o vetor após sua ordenação. Veja o exemplo:

Vetor Aleatório: 2 3 1 0 2 1 9 8 6 4

Vetor Ordenado: 0 1 1 2 2 3 4 6 8 9

7. Faça um programa em C que gere aleatoriamente uma String contendo 10 letras sem que haja repetição entre elas. Após isso solicite ao usuário que digite várias palavras até que esta seja igual a “sair”. Para cada palavra digitada verifique se com as letras contidas no vetor aleatório gerado seria possível formá-las. Veja o exemplo:

Vetor Aleatório: **AgiBalEduS**

Digite uma palavra: gil  
Resposta: **é possível**

Digite uma palavra: maria  
Resposta: **não é possível**

8. Faça um programa em C que gere aleatoriamente, através de um laço de repetição for(), os elementos (com valores entre 10 e 40) de uma matriz inteira contendo 5 linhas por 5 colunas. A matriz gerada não pode possuir elementos repetidos. Após exibir o conteúdo da matriz selecione aleatoriamente uma de suas linhas e encontre o maior elemento contido nela.

Veja o exemplo:

Matriz aleatória gerada: **10 11 12 13 14**  
**16 17 18 19 20**  
**22 23 24 25 26**  
**28 29 30 31 32**  
**33 34 35 36 37**

Linha aleatória selecionada: **3**  
Maior elemento da linha **3**: **32**

9. Faça um programa em C receba os nomes de autores de livros e converta-os para suas respectivas referências bibliográficas. Solicite os nomes até que o usuário digite a palavra "vazio". Veja o exemplo:

Digite um nome: Gil Eduardo de Andrade  
Referência: **ANDRADE, Gil E.**

Digite uma palavra: Luiz Lima  
Referência: **LIMA, Luiz**

Digite uma palavra: Marcos Alberto Vieira dos Santo e Silva  
Referência: **SILVA, Marcos A. V. S.**

10. Faça um programa em C que gere aleatoriamente, através de um laço de repetição for(), os elementos (com valores entre 10 e 50) de uma matriz inteira contendo 5 linhas por 5 colunas. A matriz gerada não pode possuir elementos repetidos. Após exibir o conteúdo da matriz ordene (linha por linha) seus elementos de forma crescente e exiba a matriz agora ordenada.

Veja o exemplo:

Matriz aleatória gerada: **28 29 30 31 32**  
**10 11 12 13 14**  
**22 23 24 25 26**  
**33 34 35 36 37**

**16 17 18 19 20**

Matriz em ordem crescente:

**10 11 12 13 14**

**16 17 18 19 20**

**22 23 24 25 26**

**28 29 30 31 32**

**33 34 35 36 37**