

### Avaliação : Lista de Exercícios

A lista pode ser realizada em equipes de até duas pessoas.

Exercícios com respostas iguais serão desconsiderados.

Entregar pseudocódigo escrito à mão e arquivo codificado na linguagem C, em mídia (pendrive, CD ou enviar por e-mail).

Assunto: Estruturas de Repetição

1) Escrever um algoritmo que gera os 30 primeiros termos da Série de Fibonacci e escreve os termos gerados com a mensagem: "é primo" ou "não é primo" conforme o caso.

2) Escrever um algoritmo que lê um número não determinado de valores para m, todos inteiros e positivos, um de cada vez.

Se m for par, verificar quantos divisores possui e escrever esta informação.

Se m for ímpar e menor do que 12 calcular e escrever o fatorial de m.

Se m for ímpar e maior ou igual a 12 calcular e escrever a soma dos inteiros de 1 até m.

3) Escrever um algoritmo que gera e escreve os 5 primeiros números perfeitos. Um número perfeito é aquele que é igual a soma dos seus divisores. (Ex.:  $6 = 1 + 2 + 3$ ;  $28 = 1 + 2 + 4 + 7 + 14$  etc.).

Assunto: Estruturas de Repetição e Vetores

4) Escrever um algoritmo que lê um vetor N[20] e o escreve. Troque, a seguir, o 1o elemento com o último, o 2o com o penúltimo, etc., até o 10o com o 11o e escreva o vetor N assim modificado.

5) Escrever um algoritmo que lê um vetor K(20) e o escreve. Troque, a seguir, os elementos de ordem ímpar com os de ordem par imediatamente seguintes e escreva o vetor k modificado.

6) Construa um algoritmo que representa a estratégia descrita abaixo para busca em um vetor já ordenado ( chama-se busca binária). O usuário entra com os valores ordenados de um vetor e um valor a pesquisar. Construa uma solução genérica e que funcione para outro vetor ordenado.

### Descrição do Funcionamento

Considere o vetor ordenado dado pelo usuário  $L = \{1, 3, 4, 6, 8, 9, 11\}$ . X é o valor que o usuário quer descobrir se existe no vetor.

Exemplo 1) O usuário entra com um valor  $X = 4$ .

O algoritmo compara X ao valor do meio do vetor (6). Se X é menor, a busca ocorre na primeira parte, antes do meio, ou seja, com  $L = 1, 3, 4$ .

Então, novamente, o algoritmo compara X ao elemento do meio do vetor (3). Se X é maior, repete com com L reduzido (4).

Compara X ao meio do vetor que restou (4). Se são iguais, achou X e é o valor achado, senão X não existe no vetor.

Exemplo 2) O usuário entra com um valor  $X = 13$ .

O algoritmo compara X com o valor do meio (6). Se X é maior, a busca ocorre na segunda parte do vetor, depois do meio, ou seja com  $L = 8, 9, 11$ . Então, novamente, o algoritmo compara X ao elemento do meio do vetor (9). Se X é maior, repete com com L reduzido ( $L = 11$ ). Compara X ao meio do vetor que restou. Se são iguais, achou X e é o valor achado, senão X não existe no vetor.

Exemplo 3) O usuário entra com um valor  $X = 6$ .

O algoritmo compara X com o valor do meio (6). Se X é igual, a busca acaba e achou o elemento.

- 7) Leia 3 vetores de 9 posições e crie outro com o 1o terço do primeiro, o segundo 3o. do segundo e o ultimo terço do 3o. Escrever o vetor resultante ao final.
- 8) Leia um vetor de 10 posições e verifique se existem valores iguais e os escreva.
- 9) Leia um vetor de 20 posições e elimine as posições com valor zero avançando uma posição, com os com os valores subsequentes do vetor. Dessa forma todos “zeros” devem ficar para as posições finais do vetor.