



Pontifícia Universidade Católica de Minas Gerais

ICEI – Instituto de Ciências Exatas e Informática

DCC – Departamento de Ciência da Computação

Campus Belo Horizonte – Unidade Coração Eucarístico

Bacharelado em Ciência da Computação

MAIOR UNIVERSIDADE CATÓLICA DO MUNDO - Fonte: Vaticano

MELHOR UNIVERSIDADE PRIVADA DO BRASIL - Guia do Estudante, por Gx

ENTRE AS MELHORES UNIVERSIDADES DO MUNDO - Times (Ranking Times High Education)

COMPUTAÇÃO PUC MINAS: SEMPRE 2º/3º LUGAR DO PAÍS (RH) – Folha de São Paulo, RUF

CIÊNCIA DA COMPUTAÇÃO PUC MINAS: SEMPRE 4 OU 5 ESTRELAS - Guia do Estudante

Algoritmos e Estruturas de Dados I

Professor: Lúcio Mauro Pereira

Lista de Exercícios nº 20

24 de abril de 2023

Arranjos Unidimensionais - Vetores

Estudar:

Ambas as obras estão disponíveis na biblioteca também de forma digital. Para ler o e-book, acesse o site da biblioteca da PUC Minas.

Obra: Fundamentos da Programação de Computadores. Autora: Ana Ascêncio

Estudar o capítulo 6 - Vetor

Obra: C: como programar. 8ed. Autor: Deitel.

Estudar o Capítulo 6: **Arrays**

Para cada problema proposto neste caderno de exercícios:

- *Elaborar um modelo de solução. Expressá-lo através de fluxograma e/ou texto estruturado – algoritmo.*
- *Codificar a solução através da linguagem C.*

Obs: **Argumentos** de uma função correspondem aos parâmetros requeridos no momento em que a função é evocada.

Para cada item desta lista, crie uma (ou mais) função(ões) que implemente(m) a solução proposta por você para cada problema. Considere o grau de reusabilidade provido por sua solução. Em seguida, construa uma função principal (main) que permita testar a solução apresentada de diferentes formas.

Busque alternativas para simplificar a interface da função utilizando recursão indireta.

1. Construa um programa que leia uma palavra pelo teclado e calcule e escreva:
 - o tamanho da palavra (número de caracteres);
 - o número de letras iguais à letra 'A' (maiúsculas ou minúsculas) presentes na palavra lida.
2. Construa um procedimento que troque o primeiro elemento do arranjo com o último.
Argumento: um arranjo de reais e um valor inteiro relativo ao tamanho do arranjo.
Teste o método criado a partir de main().
3. Construa um procedimento que troque dois elementos de duas posições quaisquer de um arranjo.
Argumentos: Um arranjo de reais, o tamanho do arranjo e dois valores inteiros relativos às duas posições a serem trocadas.
Teste o método criado a partir de main().

4. Construa uma função que verifique se um arranjo de notas dos alunos é válido ou não. O arranjo estará válido caso todos os valores nele armazenados possuem valores dentro do domínio da nota: 0..100.
A função deverá retornar *verdadeiro*, se o arranjo for válido, ou *falso*, caso contrário.
Argumentos: Um arranjo de reais e o tamanho do arranjo.
Teste o método criado a partir de main().
5. Construa uma função que calcule e retorne o número de alunos aprovados em uma disciplina. As notas dos alunos deverão estar armazenadas em um arranjo de reais. O aluno estará aprovado quando a sua nota final for igual ou superior a 60 pontos.
Argumentos: Um arranjo de reais com as notas da turma e o tamanho do arranjo.
* Utilize a função criada na questão anterior para verificar se o arranjo recebido é válido ou não.
Caso não seja válido, a função deverá gerar um valor igual a zero.
Teste o método criado a partir de main().
6. Construa uma função que receba um arranjo de inteiros e o seu tamanho e preencha-o com a série de Fibonacci. Observe que o arranjo lhe permite criar uma interessante versão iterativa (não use recursividade).
Teste sua solução a partir do método principal.