



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Pato Branco
Disciplina: Estruturas de Dados, Pesquisa e
Ordenação
Professora: Emanoeli Madalosso
Curso: Tecnologia em Análise e Desenvolvimento
de Sistemas



Avaliação 3 - 11/12/2019

- 1) **(3,0 pontos)** Neste exercício você deve criar uma função para ordenar o vetor abaixo de forma DECRESCENTE (veja que a maior parte dos itens do vetor já está na posição correta): 32, 45, 50, 18, 29, 16, 12, 8, 6, 4, 2, 1

Dentre os métodos de ordenação estudados, vimos que alguns métodos levam vantagem com vetores que já estão ordenados ou mesmo quase ordenados. Use um desses métodos.

- 2) Resolva as questões abaixo considerando busca BINÁRIA:

- a) **(1,5 pontos)** Crie uma função chamada `busca_binaria_decrescente(int vetor[], int n, int chave)`, onde:

vetor = vetor já ordenado de forma DECRESCENTE;

n = tamanho do vetor;

chave = valor a ser procurado no vetor.

A função deve retornar o índice onde a chave foi encontrada. Caso não seja encontrada, retornar -1.

- b) **(1,5 pontos)** Crie uma função chamada `busca_binaria_decrescente_2(int vetor[], int n, int chave)`, que nada mais é do que a função anterior adaptada para retornar o índice da PRIMEIRA ocorrência da chave dentro do vetor.

Veja o exemplo:

Vetor decrescente: 50, 45, 32, 29, 29, 29, 29, 18, 16, 12, 8, 6, 4, 2, 1

Item buscado: 29

Posição retornada pela função: 6

- 3) **(3,0 pontos)** O problema dos Números Maravilhosos ou Números de Granizo, intriga matemáticos. Ele consiste dos seguintes passos:

- Selecionar um número inteiro positivo n , que será a “semente” de uma sequência de números.
- Se n é par, o próximo número da sequência é $n / 2$
- Se n é ímpar, o próximo número da sequência é $3n + 1$

O fato intrigante é que para todos os números já testados, a sequência segue um padrão: independente do valor de n , ela chega em um ponto em que atinge 4, 2, 1 e então passa a repetir esses números:

$n = 3$: 10, 5, 16, 8, 4, 2, 1, 4, 2, 1, ...

$n = 4$: 2, 1, 4, 2, 1, ...

$n = 5$: 16, 8, 4, 2, 1, 4, 2, 1, ...

$n = 6$: 3, 10, 5, 16, 8, 4, 2, 1, 4, 2, 1, ...

Implemente uma função recursiva que recebe como parâmetro uma semente e imprime a sequência de números maravilhosos gerada a partir dela. Sua função deve parar de gerar a sequência ao atingir o primeiro número 1.