

Algoritmos e Estrutura de Dados I

Fabício Olivetti de França e Paulo Henrique Pisani

13 de março de 2019

Lista de Exercícios 2

Algoritmos de Busca

- 1) Altere os algoritmos de busca dado em aula para, além de buscar por um valor, atualize-o por outro.
- 2) Altere os algoritmos de busca sequencial para o uso de lista ligada.
- 3) Modifique a busca sequencial e a busca binária para retornar um ponteiro para o primeiro registro no caso de múltiplos registros com chaves iguais.
- 4) Implemente a busca de fibonacci https://en.wikipedia.org/wiki/Fibonacci_search_technique.
- 5) Verifique a seguinte afirmação: se nossa lista possui 10 ou menos elementos, a busca sequencial é mais rápida que a busca binária! Crie um programa que faça diversas buscas em uma lista de tamanho 10 e outra de tamanho 5 e compare o tempo total usando as duas abordagens. Se a afirmação for verdadeira, crie um algoritmo híbrido de busca sequencial e binária.
- 6) Adapte o algoritmo de busca binária para o caso de a array estar ordenada de forma decrescente.
- 7) O algoritmo de **interseção galopante** entre duas arrays ordenadas a e b pode ser descrito como:
 - Para cada elemento x da array a :
 - Faça uma busca binária de x em b , se bem sucedida, insira na lista de resultados.

Implemente esse algoritmo e faça testes em que a lista a é maior que a lista b e vice-versa. Teste com tamanhos e diferenças significativas em que um tem dezenas de milhares de elementos e a outra milhões de elementos.

Árvores Binárias e Árvores Binárias de Busca

- 1) Quantas árvores diferentes existem com três nós, A , B , e C ?
- 2) Verdadeiro ou falso: em um diagrama convencional de árvore (com a raiz no topo), se o nó X possui um nível maior que o nó Y , ele é desenhado abaixo de Y .
- 3) Escreva algoritmos recursivos e não-recursivos para:
 - Calcular número de nós de uma árvore binária
 - Soma dos valores de todos os nós de uma árvore binária (desafio: aplicar uma operação cumulativa de todos os nós da árvore, a função deve receber a árvore, um ponteiro pra função e um valor inicial)
 - Altura de uma árvore binária
- 4) Escreva um algoritmo que determine se uma árvore binária é completa e outro para indicar se é cheia
- 5) Escreva funções para calcular:
 - Quantidade nós folha de uma árvore binária
 - Quantidade de nós internos (não folha) de uma árvore binária
 - Imprima os níveis de uma árvore. Imprima primeiro os nós em profundidade 0, depois os nós em profundidade 1, e assim por diante.
 - Escreva uma função para verificar se uma árvore binária é uma árvore binária de busca
 - Escreva uma função para liberar uma árvore binária da memória
 - Escreva uma função de inserção de elementos em árvore que não permita a inserção de elementos repetidos
- 6) Desenhe árvores binárias que representem as seguintes expressões matemáticas:
 - a. $2(a - b/c)$
 - b. $a + b + 5c$
- 7) Verdadeiro ou falso: os nós terminais de uma árvore binária sempre aparecem na mesma ordem relativa em qualquer forma de percurso
- 8) Crie um algoritmo `reconstroi_arvore` que recebe duas listas de nós na ordem *pré-ordem* e *em-ordem* e reconstrói a árvore original. É possível fazer o mesmo com as listas *pré-ordem* e *pós-ordem*? E *em-ordem* e *pós-ordem*? Assuma que todos os nós tem um rótulo distinto.
- 9) Para o conjunto de números $\{1, 4, 5, 10, 16, 17, 21\}$, desenhe árvores binárias de busca de altura 2, 3, 4, 5, e 6.
- 10) Mostre que se um nó de uma árvore binária possui dois filhos, então seu sucessor não possui filho a esquerda e seu predecessor não possui filho a direita