

Exercícios – Estrutura de Dados II

Questão 1 – Elabore um algoritmo de pesquisa binária recursiva para encontrar a posição de um nome em uma lista ordenada alfabeticamente. A lista deve receber n nomes e receber um nome para encontrá-lo e mostrar a posição do mesmo.

Questão 2 – Calcule o tempo de complexidade do algoritmo de ordenação Insertion Sort.

```
INSERTION-SORT(A)
1  for j = 2 to A.length
2    key = A[j]
3    // Insert A[j] into the sorted
      sequence A[1 .. j - 1].
4    i = j - 1
5    while i > 0 and A[i] > key
6      A[i + 1] = A[i]
7      i = i - 1
8    A[i + 1] = key
```

- Quando que vai ocorrer o pior caso e o melhor caso?

Questão 3 – Monte a árvore da codificação de Huffman e decodifique o texto abaixo:

001110110110010011011011011111010111001111101001101111101011100110110
1111100101110111110000

z = 1

q = 1

e = 4

u = 1

f = 1

r = 2

s = 1

o = 1

$v = 1$

$' ' = 3$

Questão 4 – Você está trabalhando no desenvolvimento de um sistema para uma biblioteca. O objetivo é implementar uma tabela hash com 17 endereços base para gerenciar o acervo de livros, empregue-os com o método da divisão como função de dispersão. Em seguida, a aplicação tem os seguintes livros:

01. Clean Code: A Handbook of Agile Software Craftsmanship
02. The Pragmatic Programmer: Your Journey to Mastery
03. Design Patterns: Elements of Reusable Object-Oriented Software
04. JavaScript: The Good Parts
05. Effective Java
06. Python Crash Course: A Hands-On, Project-Based Introduction to Programming
07. Head First Design Patterns
08. Code Complete: A Practical Handbook of Software Construction

Você precisa criar uma tabela hash para armazenar esses livros de forma eficiente. Cada livro deve ser inserido na tabela usando o título como chave (quantidade totais de letras, desconsiderando o espaço).

- Escreva o conjunto das chaves envolvidas em colisões.
- Agora faça com que a tabela de dispersão trate colisões por meio de encadeamento.
- Neste sistema de biblioteca, o método da divisão foi eficiente?