UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE EXATAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
PROFA. LEILA MACIEL DE ALMEIDA E SILVA

LISTA OBRIGATÓRIA PARCIAL DE PROJETO E ANÁLISE DE ALGORITMO

SEMANAS 6 e 7

Nas resoluções das questões a seguir, inclua:

a) explicação da sua ideia de solução
b) descrição da solução em pseudo-linguagem;
c) discussão da complexidade da solução;
d) implementação do algoritmo.

1. Dados um texto *T* e um padrão *P*, de tamanhos *n* e *m*, respectivamente, encontre o índice da última ocorrência de *P* em *T*. Por exemplo,
   T: Deslumbrem-se com a beleza das araras durante o passeio.
   P: ara
   saída: 34

   (i) Adaptando o KMP;
   (ii) Adaptando o Horspool

2. Dados um texto *T* e um padrão *P*, de tamanhos *n* e *m*, respectivamente, encontre quantas vezes ocorrem todos os prefixos de *P* em *T*, adaptando o algoritmo de Rabin-Karp.
3. Uma *substring A* de uma *string B* é uma sequência de caracteres consecutivos de de *B*, de tamanho menor ou igual ao tamanho de *B*. Uma *strin*g circular, é uma *string* onde o último caractere precede o primeiro, na forma de um anel. Seja *S* uma *string* circular de tamanho *n*. Elabore um algoritmo **linear** para determinar se uma *string T* é ou não uma *substring* de *S*. Por exemplo, para a *string* circular casa, as *strings* saca e asa são substrings de casa, mas a *string* sacas não é, pois tem tamanho maior que casa, assim como a *string* saco não é, pois a letra o não ocorre em casa.

   C
   A A
   S

4. Dado um conjunto de palavras de tamanho no máximo *m*, e uma matriz quadrada de letras de ordem *n*, *n* > *m*, localize onde cada palavra ocorre na matriz. A palavra pode ocorrer em uma das linhas ou colunas da matriz, no sentido direto ou reverso. A saída de seu algoritmo deve incluir a linha e coluna da primeira e última letra da palavra. A sua solução deve necessariamente usar um dos algoritmos vistos nas semanas 6 e 7. Ex:
   Palavras: lebre, macaco, orca, pato, porco, vaca, ...
   Saída: (lebre, (1, 1), (5,1)), (macaco, (3, 2), (3,7)), (orca, (6, 11), (9,11)), (pato, (3, 11), (6,11)), (porco, (1, 7), (1,3)), (vaca, (2, 5),(2,8)), ...
   L M O C R O P B G C A

E M Y K V A C A G A L
B M A C A C O M A L P
R L L M A T N G A M A
E L E P A T N M A C T
O U R S T I F O C C O
M R U Y Y T X Z E B R
N A U T I C H Z A T C
P A M E L H E U S A A
Y L U L H A B I U S L
B C V A B E L H A B A

5. Para poder utilizar o método de Huffman integralmente, são necessárias as seguintes funções:
    a) arvoreHuffman : cria uma árvore de Huffman a partir de um texto
    b) tabelaCodigo: cria a tabela de codificação a partir da árvore de Huffman
    c) codifica: usa a tabela de código criada pela função anterior para codificar o texto informado
    d) decodifica: usa a árvore de Huffman para decodificar uma mensagem comprimida

    Nesta questão além de estruturar a solução de cada uma destas funções em pseudo-linguagem e implementar as funções, você deve integrar estas funções em um programa que dado uma mensagem exibe a mensagem codificada e em seguida a mensagem decodificada para confirmar que é igual à mensagem original.

    Ex:
    Mensagem original: ABRACADABRA
    Mensagem comprimida: 01011001111011100101100
    Mensagem recomposta: ABRACADABRA