

1) Existe a necessidade de avisar um colega de vocês que determinado professor marcou prova. Para isso, será utilizado um canal extremamente rápido mas que possui um custo de utilização consideravelmente alto, logo, é interessante que sejam utilizados dados com compactação para diminuir o custo do envio. Para fazer a compactação, utilize o Código de Huffman para as frequências:

carácter:	a	e	h	f	m	n	i	o	v	p	s	r	d
frequência:	15	10	4	2	6	3	6	8	2	1	2	1	4

Assim que construída a árvore de Huffman, codifiquem a seguinte mensagem: **“aprovafoiadiada”**.

2) Escreva um algoritmo guloso onde: A entrada é um conjunto de palavras (cadeias de caracteres quaisquer) que formam um dicionário ***D*** e uma frase ***f*** (outra cadeia de caracteres), onde todo caractere de ***f*** está em ***D***. A saída é uma sequência de segmentos de palavras que concatenados formam ***f***. Um segmento de palavra é representado por uma tripla ***(p, ini, fim)*** onde ***p*** é o número da palavra no dicionário ***D***, ***ini*** é o número do primeiro caractere do segmento e ***fim*** é o número do último caractere do segmento.

Por exemplo: ***D*** = (camelo, aguia, sapo), ***f*** = guloso;

saída: (2, 2, 3), (1, 5, 6), (3, 1, 1), (1, 6, 6).

Seu algoritmo deve gerar o mínimo de triplas possível. *Mostre* que o seu algoritmo funciona com esta característica.

3) Suponha dado um conjunto de livros numerados de 1 a n . Suponha que o livro i tem peso $p[i]$ e que $0 < p[i] < 1$ para cada i . Considere o problema de acondicionar os livros no menor número possível de envelopes de modo que cada envelope tenha no máximo 2 livros e o peso do conteúdo de cada envelope seja no máximo 1.

Escreva um algoritmo guloso $\text{Min-Env}(p, n)$ que recebe um vetor $p[1..n]$ e devolve o número mínimo de envelopes. O consumo de tempo do seu algoritmo deve ser $O(n \lg n)$. Procure mostrar que seu algoritmo está correto.
