CANA
Lista 02

Victor Campos

1. Considere uma distribuição com n possíveis resultados, com probabilidades $p_1, p_2, \ldots, p_n$.

   (a) Para esta parte da questão, assuma que cada p_i é uma potência de 2, ou seja, $1/2^k$. Suponha que realizamos o experimento m vezes e que a i-ésima saída ocorreu exatamente mp_i vezes, para todo $i \in \{1, \ldots, n\}$. Mostre que a codigicação de Huffman aplicada nesta sequência formaria uma codificação de tamanho

   $$\sum_{i=1}^{n} mp_i \log_2 \frac{1}{p_i}.$$

   (b) Agora, considere uma distribuição arbitrária, onde as probabilidades não são restritas a potências de 2. Dada a fórmula para a entropia

   $$\sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i},$$

   para qual distribuição com n resultados esta fórmula é máxima? E quando ela é mínima?

2. Um *conjunto de arestas de feedback* de um grafo $G = (V, E)$ é um subconjunto de arestas $E' \subseteq E$ que intersecta todo ciclo de G. Forneça um algoritmo eficiente para o seguinte problema.
   Entrada: grafo $G = (V, E)$, com peso w_e nas arestas
   Saída: um conjunto de arestas de feedback $E' \subseteq E$ de custo mínimo $\sum_{e \in E'} w_e$.

3. Prove as seguintes propriedades para a codificação de Huffman.

   (a) Se algum caractere ocorre com frequência maior que 2/5, então alguma codigicação tem tamanho 1.

   (b) Se todos os caracteres ocorrem com frequência inferior a 1/3, então nenhuma codificação tem tamanho 1.

4. Escreva um algoritmo eficiente para resolver o seguinte problema.
   Entrada: grafo $G = (V, E)$, peso w_e nas arestas, subconjunto de vértice $U \subset V$
   Saída: Uma árvore geradora de custo mínimo tal que os vértices de U são folhas (tem grau 1)

   Dicas:

   - A resposta não necessariamente é árvore geradora mínima de G.
   - Quando removemos os vértices de U da solução ótima, o que sobra?

5. Para um conjunto de variáveis $x_1, \ldots, x_n$, você recebe como entrada um conjunto de restrições de *igualdade*, da forma "$x_i = x_j$", e um conjunto de restrições de *desigualdade*, da forma "$x_i \neq x_j$". Escreva um algoritmo eficiente que recebe como entrada um conjunto m restrições sobre n variáveis e decide se todas as restrições podem ser satisfeitas.

6. Um restaurante tem n clientes para serem servidos. O tempo necessário para servir cada cliente i é conhecido a priori, sendo de t_i minutos. Assim, se os clientes foram servidos em ordem de i crescente, então o i-ésimo cliente irá esperar $\sum_{j=1}^{i} t_i$ minutos. Gostaríamos de minimizar o tempo total de espera, ou seja,

$$T = \sum_{i=1}^{n} (\text{tempo que o } i\text{-ésimo cliente irá esperar}).$$

Escreva um algoritmo eficiente para calcular a ordem ótimo para servir os clientes.