¡Felicitaciones! ¡Aprobaste!

Calificación recibida 100 % **Para Aprobar** 80 % o más

Ir al siguiente elemento

Problem Set #3

Calificación de la entrega más reciente: 100 %

1. Consider an alphabet with five letters, $\{a, b, c, d, e\}$, and suppose we know the frequencies $f_a = 0.32$, $f_b = 0.25$, $f_c = 0.2$, $f_d = 0.18$, and $f_e = 0.05$. What is the expected number of bits used by Huffman's coding scheme to encode a 1000-letter document?

1 / 1 punto

- (•) 2230
- ( ) 2400
- ( ) 3000
- ( ) 3450

Correcto
For example, $a = 00, b = 01, c = 10, d = 110, e = 111$.

2. Under a Huffman encoding of n symbols, how long (in terms of number of bits) can a codeword possibly be?

1 / 1 punto

- (•) $n - 1$
- ( ) n
- ( ) $\ln n$
- ( ) $\log_2 n$

Correcto
For the lower bound, take frequencies proportional to powers of 2. For the upper bound, note that the total number of merges is exactly $n - 1$.

3. Which of the following statements holds for Huffman's coding scheme?

1 / 1 punto

- (•) If the most frequent letter has frequency less than 0.33, then all letters will be coded with at least two bits.
- ( ) A letter with frequency at least 0.5 might get encoded with two or more bits.
- ( ) If a letter's frequency is at least 0.4, then the letter will certainly be coded with only one bit.
- ( ) If the most frequent letter has frequency less than 0.5, then all letters will be coded with more than one bit.

Correcto
Such a letter will endure a merge in at least two iterations: the last one (which involves all letters), and at least one previous iteration. In the penultimate iteration, if the letter has not yet endured a merge, at least one of the two other remaining subtrees has cumulative frequency at least $(1 - .33)/2 > .33$, so the letter will get merged in this iteration.

4. Which of the following is true for our dynamic programming algorithm for computing a maximum-weight independent set of a path graph? (Assume there are no ties.)

1 / 1 punto

- ( ) If a vertex is excluded from the optimal solution of a subproblem, then it is excluded from the optimal solutions of all bigger subproblems.

- [ ] As long as the input graph has at least two vertices, the algorithm never selects the minimum-weight vertex.
- [ ] The algorithm always selects the maximum-weight vertex.
- [x] If a vertex is excluded from the optimal solution of two consecutive subproblems, then it is excluded from the optimal solutions of all bigger subproblems.

Correcto
By induction, since the optimal solution to a subproblem depends only on the solutions of the previous two subproblems.

5. Recall our dynamic programming algorithm for computing the maximum-weight independent set of a path graph. Consider the following proposed extension to more general graphs. Consider an undirected graph with positive vertex weights. For a vertex v, obtain the graph $G'(v)$ by deleting v and its incident edges from G, and obtain the graph $G''(v)$ from G by deleting v, its neighbors, and all of the corresponding incident edges from G. Let $OPT(H)$ denote the value of a maximum-weight independent set of a graph H. Consider the formula $OPT(G) = \max\{OPT(G'(v)), w_v + OPT(G''(v))\}$, where v is an arbitrary vertex of G of weight w_v. Which of the following statements is true?

1 / 1 punto

- [ ] The formula is always correct in general graphs, and it leads to an efficient dynamic programming algorithm.
- [ ] The formula is correct in path graphs but is not always correct in trees.
- [x] The formula is always correct in trees, and it leads to an efficient dynamic programming algorithm.
- [ ] The formula is always correct in trees, but does not lead to an efficient dynamic programming algorithm.

Correcto
Indeed. What running time can you get?