תרגיל בית 9

חלק א' - KMP

<u>תרגיל:</u>

נתונה מחרוזת T שאורכה n≥10. תארו אלגוריתם המוצא חלוקה T = xyx כך ש-|y| ≥10 והאורך של x מקסימלי.

*הערה – תמיד קיימת חלוקה כזו, כיוון ש- x יכולה להיות ε.

דוגמא:

|y| =11

T = aabaaaaabbcbbaaaaabaa

x = aabaa

<u>פיתרון:</u>

מה יכול להיות האורך המקסימלי של x? $|x| \leq \left\lfloor \frac{(n-10)}{2} \right\rfloor = k$

- בדוגמא : $k = \left\lfloor \frac{21-10}{2} \right\rfloor$=5.

רוצים למצוא את הרישא המקסימלית של T באורך לכל היותר k שהיא גם סיפא של T.

← נריץ KMP כאשר התבנית היא T_k והטקסט הוא k התווים האחרונים ב-T.

סיבוכיות: O(n).

P T

T = aabaadaabbccccbbacaaabaa

דוגמא –

c a a a b a a

| | | | |

a a b a a d a

|x| = 5

חלק ב' – Huffman coding, Kraft inequality

The Kraft inequality[5] tells us whether it is possible to construct a prefix-free code for a given source alphabet $\mathcal{X}$ with a given set of codeword lengths $\{l(x), x \in \mathcal{X}\}$.

Theorem 3.1 (Kraft inequality for prefix-free codes) *Every prefix-free code for an alphabet $\mathcal{X}$ with codeword lengths $\{l(x), x \in \mathcal{X}\}$ satisfies*

$$\sum_{x \in \mathcal{X}} 2^{-l(x)} \leq 1. \tag{1}$$

Conversely, if (1) is satisfied, then a prefix-free code with lengths $\{l(x)\}$ exists.

Moreover, every full prefix-free code satisfies (1) with equality and every non-full prefix-free code satisfies it with strict inequality.

Proposition 3.1 *In any optimal code tree for a prefix-free code, each node has either zero or two children.*

To see why, suppose an optimal code tree has a node with one child. If we take that node and move it up one level to its parent, we will have reduced the expected code length, and the code will remain decodable. Hence, the original tree was not optimal, a contradiction.

Proposition 3.2 *In the code tree for a Huffman code, no node has exactly one child.*

To see why, note that we always combine the two lowest-probability nodes into a single one, which means that in the code tree, each internal node (i.e., non-leaf node) comes from two combined nodes (either internal nodes themselves, or original symbols).

Proposition 3.3 *There exists an optimal code in which the two least-probable symbols:*

- *have the longest length, and*
- *are siblings, i.e., their codewords differ in exactly the one bit (the last one).*

Questions:

1. **Bad Huffman Codes**: Which of these codes cannot be Huffman codes for any probability assignment?
   (a) $\{0, 10, 11\}$.
   Solution: $\{0, 10, 11\}$ is a Huffman code for the distribution $(1/2, 1/4, 1/4)$.
   (b) $\{00, 01, 10, 110\}$.
   Solution: $\{00, 01, 10, 110\}$ is not a Huffman code because there is a unique longest codeword.
   (c) $\{01, 10\}$.
   Solution: The code $\{01, 10\}$ can be shortened to $\{0, 1\}$ without losing its instantaneous property, and therefore is not optimal and not a Huffman code.
2. **Huffman's algorithm**: Let $p_1 > p_2 > p_3 > p_4$ be the symbol probabilities for a source alphabet size $M = |\mathcal{X}| = 4$.
   (a) What are the possible sets of codeword lengths $\{l_1, l_2, l_3, l_4\}$ for a Huffman code for the given type of source?
   Solution: $\{(1, 2, 3, 3), (2, 2, 2, 2)\}$ by Kraft's inequality.
   (b) Suppose that $p_1 > p_3 + p_4$. What are the possible sets of codeword lengths now?
   Solution: Note that $p_1 > p_3 + p_4$ means that at the second stage of the Huffman algorithm, p_2 will merge with the node $p_3 + p_4$ (combined at the first stage). So $l_1 = 1$ and the codeword lengths are $\{1, 2, 3, 3\}$.
   (c) What are the possible sets of codeword lengths if $p_1 < p_3 + p_4$?
   Solution: Similar argument for $p_1 < p_3 + p_4$. At the second stage p_1 will merge with p_2 (since they are now the two lowest probabilities). In this case, the lengths will be $\{2, 2, 2, 2\}$.