PAL labs 7

2 / 11 / 2022

Let's assume that each element of a Gray code G^n, which is an $n-$tuple of zeros and ones, is stored in an array of length n. Write a pseudocode of a recursive procedure which generates the whole Gray code G^n given n.

There is an eight-element sequence $P = (000, 001, 011, 010, 110, 111, 101, 100)$ which corresponds to the Gray code G^3. We call two finite sequences A and B to be equivalent iff:

1. We get B by reversing the order of elements in A.
2. We get B by rotating the sequence A to one or more positions to the left or right.
3. There is a sequence C which is equivalent to A and B.

Find an eight-element sequence Q which is a Gray code, but which is not equivalent with P. Recall that Gray code is each binary system in which two neighboring elements differ in exactly one position.

DFA

σ :

Let's have two languages $L1$ and $L2$ over the alphabet $\{0, 1\}$. Words of $L1$ correspond to the regular expression $0^*1^*0^*1^*0^*$, and words of $L2$ correspond to the regular expression $(01 + 10)^*$.

1. Find the shortest nonempty word in the intersection of $L1 \cap L2$.
2. Find the longest word in the intersection of $L1 \cap L2$.
3. Find the shortest word which belongs to $L1$, but does not belong to $L2$.
4. Find the shortest word which belongs to $L2$, but does not belong to $L1$.
5. Find the shortest word which does not belong to the union of $L1 \cup L2$.

6/*2. Design a finite state automaton accepting exactly all worlds over the alphabet $\{0, 1\}$ which

1. contain the subsequence 1010 at least once
2. do not contain the subsequence 1010
3. contain the subsequence 1010 exactly once
4. contain the subsequence 1010 at most twice

Find a regular expression for a language over the alphabet $\{0, 1\}$,

1. whose words contain only zeros,
2. whose each word contains exactly a single one,
3. whose each word contains at least one one,
4. whose each word contains at least two ones,
5. whose each word contains even number of occurrences of ones,
6. whose each word contains odd number of ones.

Design an NFA over the alphabet $\{0, 1, 2\}$ which finds all substrings containing three zeros and two ones within a text.

Design an NFA over the alphabet $\{a, b, c, d\}$ which finds all strings of the pattern $\#ba\#\#b\#$ in a text, where the symbol $\#$ represents an arbitrary letter from the set $\{a, b, d\}$. The automaton has to be applicable to the text of arbitrary length, and ends in an acceptance state after reading the last element of the given pattern.

Design an automaton working over the alphabet A which finds all the words represented by the regular expression R.
$A = \{a, b, c\}$, $R = (ac^* + bb)^* a$