ECE374 Assignment 1

Update by jie W
01/29/2023

01/26/2023

Group & netid

Chen Si *chensi3*

Jie Wang *jiew5*

Shitian Yang *sy39*

Problem 2

2. For each of the following problems:

   i. Formulate the problem as a *regular* language (give an example of the problem instances and how they are encoded, you don't have to write every problem instance).

   ii. Describe the regular expression that describes the expression

   Note that how you encode the language matters for the regular expression you end up with.

   a Checking whether (or not) a number is divisible by 4). You are given a binary number and need to output if this number is divisible by 4.

   b The game of TicTacToe. You are given a completed tic-tac-toe board and you need to determine who won. (this won't have a clean regular expression. Just define some encoding and describe how you would build the expression, you don't need to write the whole expression out.) Hint: think about how many games of TicTacToe there are.

Answer:

(a) Since the binary numbers divisible by 4 all have 00 as their last two digits, we might define the regular language L_A as:

Base Case:

(1) $00 \in L_A$. single '0' $\notin L_A$.(we don't treat 00==0 . because it's string, not integer)

Inductive Step:

(2) $0w \in L_A$, if $w \in L_A$

(3) $1w \in L_A$, if $w \in L_A$

For example, the binary expression of 52 is 10100, and we could construct it by:

00 → Rule (1)

100 → Rule (3)

0100 → Rule (2)

10100 → Rule (3)

Therefore, we could obtain a regular expression of the language as $(0+1)^*00$ +0

Update:

Since the question 2-a requires output. we extend LA to language LA', whose alphabet is {0, 1, '10', '11'}, where

- '10' means output is 0, not a number that is divisible by 4
- '11' means output is 1, a number that is divisible by 4

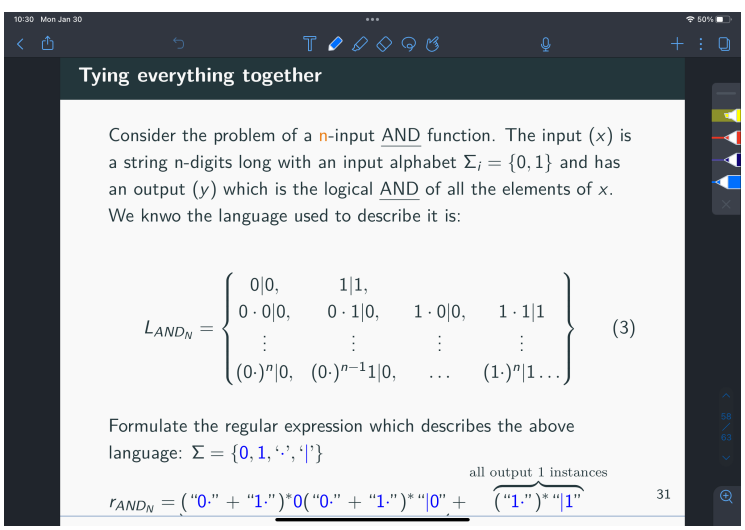
Therefore, we have the LA' defined by regular expression:

$$0+(0+1)^*00\,|\,1+(0+1)^*01\,|\,0+(0+1)^*10\,|\,0+(0+1)^*11\,|\,0$$

The operation uses only union, concatenation & Kleen star,

So the LA' is regular. and it's the one fits for 2-a

reference: Lec2 's last question



10:30 Mon Jan 30 — 60%

Tying everything together

Consider the problem of a n-input AND function. The input (x) is a string n-digits long with an input alphabet $\Sigma_i = \{0, 1\}$ and has an output (y) which is the logical AND of all the elements of x. We knwo the language used to describe it is:

$$L_{AND_N} = \left\{ \begin{matrix} 0|0, & 1|1, & & \\ 0\cdot 0|0, & 0\cdot 1|0, & 1\cdot 0|0, & 1\cdot 1|1 \\ \vdots & \vdots & \vdots & \vdots \\ (0\cdot)^n|0, & (0\cdot)^{n-1}1|0, & \dots & (1\cdot)^n|1\dots \end{matrix} \right\} \quad (3)$$

Formulate the regular expression which describes the above language: $\Sigma = \{0, 1, \text{`}\cdot\text{'}, \text{`}|\text{'}\}$

$r_{AND_N} = (\text{"0}\cdot\text{"} + \text{"1}\cdot\text{"})^*0(\text{"0}\cdot\text{"} + \text{"1}\cdot\text{"})^*\text{"}|0\text{"} + \overbrace{(\text{"1}\cdot\text{"})^*\text{"}|1\text{"}}^{\text{all output 1 instances}}$

31

(b)

There should be 8 scenarios for a player to win in a tic-tac-toe game (if it's not an unreachable game e.g. # of x - # of o > 1 or # of x - # of o < 0, we do not consider these scenarios in this problem): 3 row win scenarios, 3 column win scenarios, and 2 wins in diagonals. After defining a language which includes all strings representing the winning scenarios, we simply need to find the regular expressions of strings satisfying these winning scenarios.

When encoding the board to a string, we use the following sequence on the board to form the string:

1	2	3
4	5	6
7	8	9

Using 0 to represent empty, 1 to represent "x", 2 to represent "o".

So, the 8 situations could be denoted as:

Row wins:

$aaa(0+1+2)\ (0+1+2)\ (0+1+2)\ (0+1+2)\ (0+1+2)\ (0+1+2),$
$(0+1+2)\ (0+1+2)\ (0+1+2)aaa(0+1+2)\ (0+1+2)\ (0+1+2),$
$(0+1+2)\ (0+1+2)\ (0+1+2)\ (0+1+2)\ (0+1+2)\ (0+1+2)aaa$

Columns Wins:

$a(0+1+2)\ (0+1+2)a(0+1+2)\ (0+1+2)a(0+1+2)\ (0+1+2),$
$(0+1+2)a(0+1+2)\ (0+1+2)a(0+1+2)\ (0+1+2)a(0+1+2),$
$(0+1+2)\ (0+1+2)a(0+1+2)\ (0+1+2)a(0+1+2)\ (0+1+2)a$

Diagonal Wins:

$a(0+1+2)\ (0+1+2)\ (0+1+2)a(0+1+2)\ (0+1+2)\ (0+1+2)a,$
$(0+1+2)\ (0+1+2)a(0+1+2)a(0+1+2)a(0+1+2)\ (0+1+2)$

where a is 1 or 2 in one expression.

To judge who win, we just need to find the string represent the board belongs to which situation in those 8 situations, and the winner is the person whose mark represents the a in the situation.