National Yang Ming Chiao Tung University
Department of Electrical Engineering
Data Structures, Fall 2022

November 11, 2022
Instructor: Mark P.-H. Lin
TA: Yang-Ching Lee

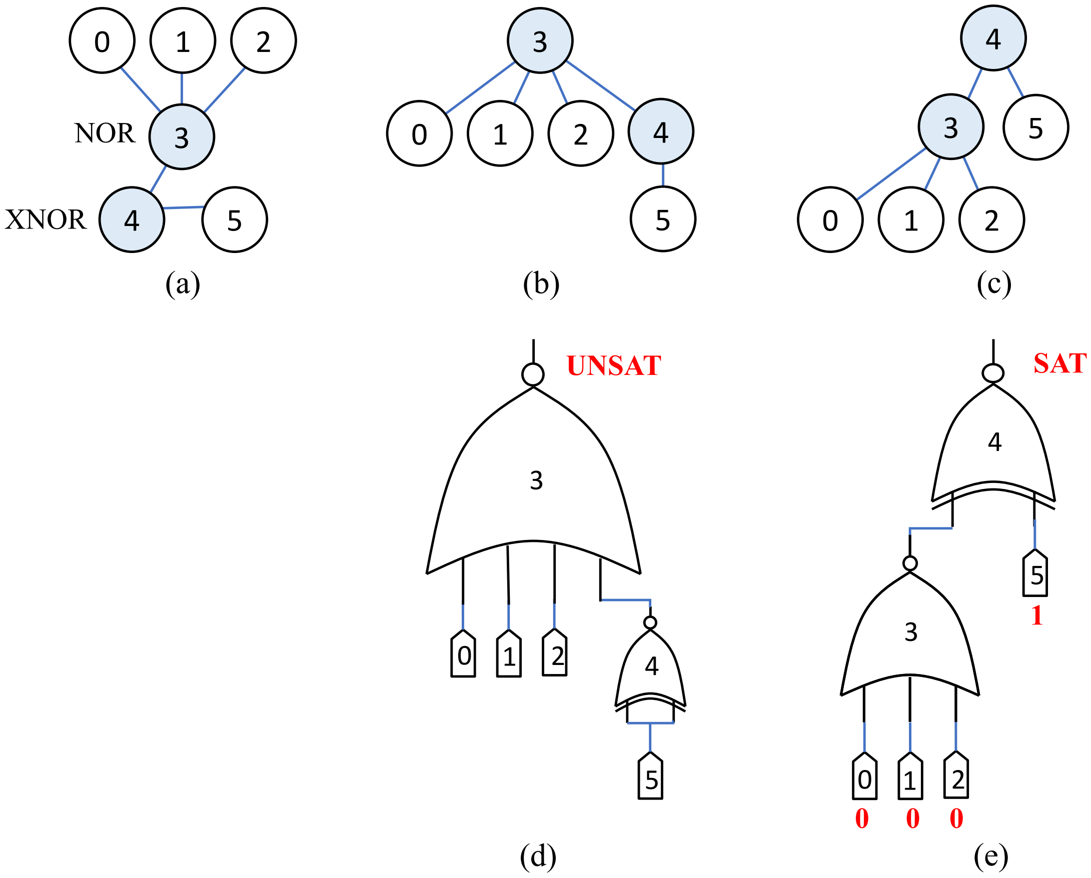
Programming Assignment #4

Trees

1 Problem Description

The "tree" data structure is one of the most widely used abstract data types that represents hierarchical relationships with a set of connected nodes. Each node in the tree can be connected to many children, depending on the type of tree, but must be connected to exactly one parent, except for the root node. There is no cycle or loop, which means that no node can be its own ancestor.

Given a set of N nodes, with the labels from 0 to $N-1$, connected by a set of of $N-1$ undirected edges without cycles, where each edge $(n_i,\ n_j)$ indicates the connection between two nodes, n_i and n_j, you can choose any node as the root, and the resulting rooted tree has height, h. Among all possible rooted trees, those with the minimum height are called minimum height trees. In this programming assignment, you are first asked to find all the minimum height trees from the N connected nodes. The figures below gives an example of the minimum height trees, as shown in Figures (b) and (c), which result from the six connected nodes in Figure (a).



(a) (b) (c)

(d) (e)

Based on the resulting minimum height trees, you are also asked to solve the Boolean satisfiability (SAT) problem. Given by the input file, each node, except those degree less than 2, is associated with either one of the following logic gate: (1) AND, (2) OR, (3) NAND, (4) NOR, (5) XOR, and (6) XNOR. In Figure (a), the nodes, n_3 and n_4, are associated with a NOR gate and an XNOR gate, respectively, while the nodes, n_0, n_1, n_2, and n_5, are not associated with any logic gate because their degrees are less than 2. Based on the resulting minimum height trees in Figures (b) and (c), the corresponding logic networks are shown in Figures (d) and (e), respectively. It should be noted that the leaf tree nodes correspond to different primary inputs. If a non-leaf tree node has only one child, such as n_4 in Figure (b), it will correspond to a 2-input logic gate, and both inputs connect to the same input signal, as shown in Figure (d).

You need to evaluate whether these logic networks are satisfiable by finding an input combination of Boolean values leading to “1” at the output. If the logic network is unsatisfiable, as seen in Figure (d), your program will output “UNSAT”. Otherwise, your program will output “SAT” together with a valid Boolean combination, as shown in Figure (e).

2 Input Format

The input file for the example in Figure (a) is given below. To simply the problem, we set total node number N to the range: $4 \leq N \leq 10$, and set the maximum degree of tree to 4.

Sample Input	Comments
6	// total node number
(0, 3)	// edge 1
(1, 3)	// edge 2
(2, 3)	// edge 3
(4, 3)	// edge 4
(5, 4)	// edge 5
3: NOR	// n_3 is associated with a NOR gate.
4: XNOR	// n_4 is associated with an XNOR gate.

3 Output Format

The output file generated by your program must follow the format below and include all the required information.

Sample Output	Comments
root = 3	// The root node of the 1^{st} minimum height tree
UNSAT	// The minimum height tree rooted at n_3 is unsatisfiable.
root = 4	// The root node of the 2^{nd} minimum height tree
SAT	// The minimum height tree rooted at n_4 is satisfiable.
0: 0	// Boolean value of n_0
1: 0	// Boolean value of n_1
2: 0	// Boolean value of n_2
5: 1	// Boolean value of n_5

4 Command-line Parameter

In order to test your program, you are asked to add the following command-line parameters to your program:

[executable file name] [input file name] [output file name]

5 Submission Information

1. Your program must be written in the C/C++ language and can be compiled on the Linux platform.
2. The source files of your program must be named with "[your student ID].h" and "[your student ID].cpp".
3. To submit your program, please archive all source files of your program into a single zip file, named "[your student ID].zip", and upload it to E3.

6 Due Date

Be sure to upload the zip file by "Wednesday, November 24, 2022". There will be a 25% penalty per day for late submissions.

7 Grading Policy

The programming assignment will be graded based on the following rules:

- Pass all sample inputs on E3 with compilable source code (50%)
- Pass five hidden test cases (50%)

The submitted source codes, which are either copied from or copied by others, will NOT be graded.