

# CS 1511 Homework 12

Mathew Varughese, Justin Kramer, Zach Smith

Friday, March 1

## 22. a)

Each  $m$  logical operation will become a gate. The  $n$  variables will become a line of inputs into the circuit. The precedence of the logical operations will be shown by the depth of the gates. When a gate is involved in an operation after a previous operation, it will move down one depth in the circuit. At the end of the circuit, The gates will compute the same answers as the boolean function in all cases.

## 22. b)

Each output of each gate in the circuit will become its own variable. We will have a program that transforms these outputs in variables. Then the output of the next gate will create a new variable that encapsulates the boolean formulas taking place at each gate. This will build until at the end of the program, we have one boolean expression that contains  $S$  logical operations. When we build our boolean function  $G$ , we need to create an XNOR for each variable in the program with the boolean function that it represents. This will create a maximum of  $S^2$  logical operations and if each XNOR is satisfied as well as the first large boolean function that our program provided, then  $G$  is satisfiable.

## 22. c)

The program that we built in the previous problem is the answer to this question. This program contains  $S$ -lines for each gate of the circuit, creating a new variable for each output of each gate, building a larger and larger boolean function. The final line of this program will be the final output of the final gate, which is one boolean function. In the other direction, the confirmed  $S$ -line program can have each variable assignment become a gate in the circuit. These gates will be combined in the order of the variable assignments.

## 23.

If you use the HALTING problem, which is in P/Poly, and set an exponential time limit to it, then it becomes a decidable language in P/Poly that is not in P. Since the program can run for exponential time, it is clearly not inside of P. The language can become decidable in P/Poly if you make all the possible inputs as unary, therefore making each having a unique size. We can then hard-wire in our circuit an answer for each of these inputs, and make the problem decidable and in P/Poly.