Introduction to Digital Computers - Spring 2001

Assignment No. 1

Course homepage: http://hyde.eng.tau.ac.il/~guy/Digital_Computers01/

Firm Deadline: March 15th - before the beginning of the lecture

1. A path in a circuit is a sequence of gates $G_0, G_1, \ldots, G_p$ such that an input of gate G_{i+1} is fed by a net which is fed by an output of gate G_i. A cycle is a closed path, namely, a path in which $G_0 = G_p$. Prove that a combinational circuit does not contain cycles.

2. Write an algorithm that is given as input a circuit and outputs whether it is a combinational circuit. If the circuit is a combinational circuit, then the algorithm outputs the linear order of the gates in the circuit $G_1, G_2, \ldots$ (according to the definition of a combinational circuit). Please write the algorithm on paper in a natural language (not a programming language). Prove the correctness of your algorithm.

3. (*suggested by Arnon Warshavsky*). Consider computing Boolean functions using a spreadsheet application (i.e. Excel). The value stored in each cell should be a Boolean value. The value of a cell may either be constant or determined by an expression whose parameters are values of other cells. For simplicity, assume that the expressions contain only the Boolean operators: not, or, and.

   - Assuming we would like the values of the cells to be well defined, define a "legal" spreadsheet. Try a few examples of illegal spreadsheets to see if your definition is used in the application.
   - How do spreadsheet applications guarantee the legality of a spreadsheet?

4. Write a (recursive) algorithm that is given as input a combinational circuit and outputs the propagation delay of the circuit. Prove the correctness of your algorithm. (You may use the algorithm from Question 2 as a procedure).

5. Prove that all OR-tree(n) circuits have the same cost. We assume that:

   (a) Every input is connected to exactly one input of an OR-gate.

   (b) Only OR-gates are used and they are connected in a tree structure.

   What is the minimal delay of an OR-tree(n)?

   Can you derive from the argument above a more general claim about the cost and minimal delay of combinatorial circuits? You can assume that all gates are binary and have the same cost and delay.