

Introduction to Computer Science E13 – Discussion Sections 1 – Week 36

1. Divide into groups of four (or three) people. One person will choose five cards to give to the first “performer”, the first performer will give four of them to the second “performer”, one at a time, and the second performer will announce what the fifth card is. Each person should practice each “performer” part at least three times.
2. Propose at least one method for improving the magic trick. For example, if the first card determines the suit, after seeing the trick repeated several times, the audience might find it easier to guess how it is done. Define an algorithm for a modification of the trick which makes this harder to see.
3. Propose an algorithm for doing a magic trick, where one performer is thinking of a number between 1 and 24, tells the audience the number, gets some cards from the audience, and passes some cards to the second performer, who announces the number.
4. Discuss how to extend this to larger numbers than 24. When might it be better to only use the color on the card (whether it is red or black, but also consider using the four different suits), rather than some permutation (ordering) of the cards?
5. Find a “bad” pair of integers for the greatest common divisor algorithm, where a pair is bad if the algorithms must perform a lot of steps relative to how large the numbers are. (One expects more steps for larger numbers.)
6. Design a circuit, using only AND, OR and NOT gates which takes three bits as input and outputs a 1 if the input has at least two zeros, and a 0 otherwise.

7. Design and draw a circuit containing only AND and XOR gates (with at most two inputs) which takes three bits as input and outputs a 1 if the input has at least two ones, and a 0 otherwise.

(In the student resources for the course textbook, under the Activities for Chapter 1, there is a simulator for logic circuits which you could use to check your circuit. It is time consuming to use, though.) As an extra challenge, try to do it so that there is only one AND gate, though more XOR gates. (Minimizing the number of AND gates can be useful in some cryptographic applications.)

8. Write up an algorithm for one of the possibilities mentioned in the first paragraph of section 0.1. (For example, give a clear, unambiguous algorithm for getting from where you live to U91 at SDU.)