Institut for Matematik og Datalogi
Syddansk Universitet

Project — Computer Science 2013

This is your "studiestartsopgave". This project is based on your courses DM534, DM535, and DM536, though most directly on DM534 since it overlaps with both of the other two. It covers stating algorithms precisely and being able to follow the execution of an algorithm (as is required in programming) and circuit design (Boolean algebra and logic). The project will be graded on a Pass/Fail basis.

The assignment is due at 23:59 on Sunday, September 22. You may write this either in Danish or English. It must be made in LaTeX, though you may draw some diagrams by hand, scan them in to make PDF files, and then include the PDF files in your LaTeX code. Write your full name, your section number, and your "instruktor"s name (Uffe Thorsen or Magnus Gausdal Find) clearly on the first page of your assignment (on the top, if it's not a cover page). You should turn in both a paper copy of this project (to Uffe Thorsen's mailbox, which is among all the "instruktor"s mailboxes) and an electronic version as a PDF file via Blackboard through your DM534 course. The assignment hand-in is in the menu for the course and is called "SDU Assignment". Keep the receipt it gives you proving that you turned your assignment in on time. **Blackboard will not allow you to turn in an assignment late.** (Ask for help early if you need help for submitting.)

You will be able to pick up your graded project in class on Thursday, October 3.

Cheating on this project is viewed as cheating on an exam. You are allowed to talk about course material with your fellow students, but working together on this project is cheating. If you have questions about the project, come to Joan Boyar or your "instruktor" for DM534.

Please note that this assignment is a compulsory part of your first-year examination. If you fail to hand in the assignment on time, you will be called in to talk with an administrator and may not be able to continue your studies.

The Project

Do the following problems and write your solutions in LaTeX. (Do not include the statements of the problems or other information not asked for in the problems.)

1. Write down both algorithms for the "magic trick" described in class in DM534 (and practiced in discussion section), where one performer passes four of five playing cards to the other, so that the other can tell what the fifth card is. (Note that you must use the algorithm presented in class, not one you know of or make up yourself.) There should be two algorithms, one for each of the two performers. Write these as algorithms, so that all steps are completely specified, without ambiguity. (Do not explain why the algorithm works, just what the algorithm is.)

2. The version of Figure 0.2 from the textbook which was presented on the slides for DM534 is as follows:

   GCD(M, N):
   { Input: two positive integers M, N }
   { Output: gcd(M, N) }

   $A \leftarrow \max(M, N)$
   $B \leftarrow \min(M, N)$

   $Q \leftarrow A$ div B
   $R \leftarrow A - (Q \cdot B)$
   while $R \neq 0$ **do**
   $A \leftarrow B$
   $B \leftarrow R$
   $Q \leftarrow A$ div B
   $R \leftarrow A - (Q \cdot B)$
   return(B)

   Suppose that as input to this is $M = 36$ and $N = 44$. Show the sequence of values for A and B that are computed by the algorithm and the result.

3. Either do the first two problems below or the third one. The third one is somewhat more challenging. In all cases, explain your solutions. The circuits themselves may be drawn neatly by hand, scanned in to make PDF files, and included in your LaTeX code as PDF files.

   (a) Design and draw a circuit containing only AND, OR and NOT gates (each gate having at most two inputs) which takes three bits as input and outputs a 1 if the input is 100, 111, 011 or 110, and a 0 otherwise. (In the student resources for the DM526 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.)

   (b) Design and draw a circuit containing only AND, OR and NOT gates (each gate having at most two inputs) which takes four bits as input and outputs a 1 if the input is 1011, 0111 or 0110, and a 0 otherwise.

   (c) Design and draw a circuit containing only AND and XOR gates (each gate having two inputs) which takes seven bits as input and outputs a 1 if the input has at least four ones, and a 0 otherwise. Use only four AND gates. How many XOR gates do you need? Hint: Look at (and use) the problem from the discussion section where you were asked to minimize the number of AND gates. Then consider how to represent in binary how many ones there are in a set of three bits. Then add two numbers, each of which has two bits, plus one extra bit.

4. Include your LaTeX code for this assignment at the end.