**German University in Cairo**
**Faculty of Media Engineering and Technology**
**Prof. Dr. Slim Abdennadher**
**Dr. Aysha ElSafty**

**Introduction to Computer Science**, Winter Semester 2017
**Practice Assignment 11**

Discussion: 30.12.2017 - 04.01.2018

**Exercise 11-1** To be Discussed in Tutorial

Simplify the Boolean expressions to a minimum number of literals using the Boolean algebra. Please mention the applied rules.

| | | |
|---|---|---|
| $x + 0 = x$ | $x * 1 = x$ | |
| $x + 1 = 1$ | $x * 0 = 0$ | |
| $x + x = x$ | $x * x = x$ | |
| $x + x' = 1$ | $x * x' = 0$ | |
| $(x')' = x$ | | |
| $x + y = y + x$ | $xy = yx$ | Commutativity |
| $x + (y + z) = (x + y) + z$ | $x(yz) = (xy)z$ | Associativity |
| $x(y + z) = xy + xz$ | $x + yz = (x + y)(x + z)$ | Distributivity |
| $(x + y)' = x'y'$ | $(xy)' = x' + y'$ | DeMorgan's Law |

a) $ABC + ABC' + A'B$

b) $(A + B)'(A' + B')$

c) $(A + B' + AB')(AB + A'C + BC)$

d) $P'XY + PX'Y + PXY' + PXY$

e) $(AB)'(A + B)$

f) $B + A'C + AB'$

g) $AB + A'C + BC$

**Exercise 11-2**

Given the following Boolean expression, simplify it to a minimum number of literals using the Boolean algebra. Please mention the applied rules.

$$((A + B)(B' + C' + D')) + B'C'(A + B' + C) + A'C + D$$

**Hint:** The circuit of the simplified expression consists of zero gates.

**Exercise 11-3**

Use AND, OR and NOT gates to implement the circuits represented by the following two expressions:

$$\begin{aligned} S &= P'X'Y + P'XY' + PX'Y' + PXY \\ C &= P'XY + PX'Y + PXY' + PXY \end{aligned}$$

**Exercise 11-4** To be Discussed in Tutorial

Draw a logic circuit that corresponds to each of the expressions shown below:

a) $AB' + A'C'D' + A'B'D + A'B'CD'$

b) $B' + A'C'D'$

c) $(A' + B' + C + D')(A + B + C' + D)$

**Exercise 11-5**

Given the following the following truth table, where **A**, **B** and **C** are the input variables and **X** is the output variable.

| **A** | **B** | **C** | **X** |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

a) Use the sum-of-products algorithm to find the Boolean expression that describes the output of the truth table.

b) What is the functionality of the circuit?

c) Draw the Boolean circuit. **Note** that each gate can have only two inputs.

**Exercise 11-6** Comparator

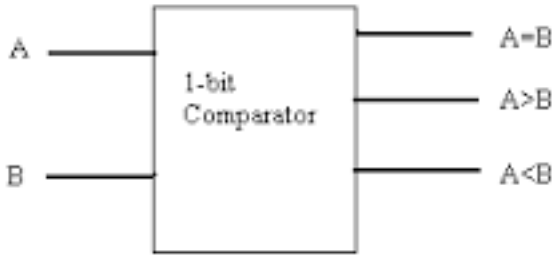
A one-bit comparator is a circuit that takes two numbers consisting of one bit each and outputs 1 if the numbers are equal, 0 otherwise.

a) Construct a truth table for a one bit equality comparator.

b) Assume that you have already manufactured one-bit comparators.


Design a circuit that uses one-bit comparators and AND-gates to check the equality of two numbers consisting of 4 bits each.

c) Assume that our one-bit comparator was modified to have two input variables $A, B$ and three output variables (one checking for $A = B$, one checking for $A > B$ and one checking for $A < B$).




Design a circuit that uses the modified one-bit comparators with other gates to compare two numbers consisting of 2 bits each. **Do not draw the truth table.**

**Exercise 11-7** To be Discussed in Tutorial

A circuit should be designed to perform the operation $(A - 1)$ where $A$ represents a number in sign/magnitude notation consisting of 2 bits.

a) How many output variables are needed? Justify your answer.

b) Construct the truth table for this circuit.

c) Use the sum-of-products algorithm to find the Boolean expressions that corresponds to the truth table.

d) Simplify the Boolean expressions that you got in c) to a minimum number of literals using the Boolean algebra. Please mention the applied rules.

e) Draw a logic circuit that corresponds to the simplified expressions you got in d).