General Computer Science I (320101) Fall 2011
Assignment 11: Propositional Logic and Quine McCluskey
(Given Dec. 2., Due Dec. 8.)

20pt

Problem 11.1 (Propositional Logic with Hilbert Calculus)

- Give an example for each of the following:
  1. a Boolean expression which is falsifiable, but also satisfiable
  2. a Boolean expression which is unsatisfiable
- Consider the Hilbert-style calculus given by the axioms
  1. $K := P \Rightarrow Q \Rightarrow P$
  2. $S := (P \Rightarrow Q \Rightarrow R) \Rightarrow (P \Rightarrow Q) \Rightarrow P \Rightarrow R$

  and the rules:
  1. $\dfrac{\mathbf{A} \Rightarrow \mathbf{B} \quad \mathbf{A}}{\mathbf{B}}$MP
  2. $\dfrac{\mathbf{A}}{[\mathbf{B}/X](\mathbf{A})}$Subst

  Prove that $(\mathbf{C} \Rightarrow \mathbf{C}) \Rightarrow (\mathbf{C} \Rightarrow \mathbf{C})$.

80pt

Problem 11.2 (Implementing Quine-McCluskey)
Implement the Quine-McCluskey algorithm in SML. The function `QuineMcCluskey` should take a truth table of type `(int list * int) list` as input and return a minimal covering prime implicant polynomial as a `string`.

Please follow the steps described in the slides and mark them clearly in your code. Verify your solution for the first problem using this algorithm.

Example:

```
- QuineMcCluskey([([0,0], 1),([0,1], 0),([1,0], 0),([1,1], 1)])
val it = "(-x1)(-x2) + x1x2" : string
```

Hint: Keep in mind that you have to output only one solution. Solve problem 3 first! You will see that, finding the essential prime implicant resumes to finding which ones can be non-essential, and also that, no matter how you choose between these non-essential prime implicants, you will still get a correct solution.

Problem 11.3 (QMC application) 20pt

Execute the Quine-McCluskey algorithm to get the minimum polynomial for the function with the provided truth table:

x_0	x_1	x_2	x_3	f
F	F	F	F	T
F	F	F	T	T
F	F	T	F	T
F	F	T	T	F
F	T	F	F	F
F	T	F	T	T
F	T	T	F	T
F	T	T	T	T
T	F	F	F	F
T	F	F	T	F
T	F	T	F	F
T	F	T	T	F
T	T	F	F	T
T	T	F	T	F
T	T	T	F	T
T	T	T	T	F