Course on Formal Methods 2010-2011

Logic & Theorem Proving

10 February 2011

1. Given the formula $((x \vee y) \Rightarrow z) \Rightarrow (x \wedge y)$, give (if they exist) two assignments to variables x, y, z that make the formula true and two other assignments that make it false.

2. Transform the formula $(x \Rightarrow y) \Rightarrow (\neg y \Rightarrow \neg z)$ into CNF.

3. Prove $((A \vee B) \Rightarrow C) \Rightarrow ((D \Rightarrow A) \Rightarrow ((B \vee D) \Rightarrow C))$ using natural deduction by indicating the rule applied at each step.

4. Prove $(\exists x.(P\,x)) \Rightarrow ((\forall x.((P\,x) \Rightarrow (Q\,x\,x))) \Rightarrow (\exists x.(\exists y.(Q\,x\,y))))$ using natural deduction by indicating the rule applied at each step.

5. Given the λ-expression $t = (\lambda xy.yw(\lambda x.yx))a(\lambda w.wx)$, mark each variable occurrence in t as binding, bound or free. For each bound occurrence, indicate the corresponding binding occurrence. Then, apply β-reduction to t by underlining the redex at each step.

6. Let $\Sigma = \{f : \sigma \to \sigma \to \tau \to \rho,\ g : \tau \to \tau \to \sigma\}$ and $\Gamma = \{x : \sigma, y : \tau\}$. Derive a type judgement for the term $f\,x\,(g\,y\,y)$.

7. A *boolean expression* over any type T of elements is either the constant *true* or a variable denoted by an element in T or the negation of a boolean expression over T or the conjunction of two boolean expressions over T. Give a definition of the type `BoolExp` using the derived rule for data type definition, discuss its characteristics and finally give an example of a term of the defined data type.