Course on Formal Methods 2010-2011

Logic & Theorem Proving

24 February 2011

1. Given the formula $((x \wedge y) \Rightarrow \neg z) \Rightarrow (x \vee y \vee z)$, 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 $\neg((x \Rightarrow \neg y) \wedge (\neg x \Rightarrow \neg z))$ into CNF.

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

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

5. Given the λ-expression $t = (\lambda xyz.xy(xa)z)(\lambda y.zyx)$, 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 $\Sigma = \{f : \sigma \rightarrow \tau \rightarrow \rho,\ h : \tau \rightarrow \tau\}$ and $\Gamma = \{x : \sigma, y : \tau\}$. Derive a type judgement for the term $\lambda x^{\sigma}.\, f\, x\, (h\, y)$.

7. A *stack* over any type T of elements is either the empty stack or an element of T pushed on a stack over T. Give a definition of the type `Stack` using the derived rule for data type definition, discuss its characteristics and finally give an example of a term of the defined data type.