1 Computational Logic 2008 - Dr G.Bellin

Coursework 5

1. Let ⊥ be a propositional constant interpreted as a logically absurd sentence. The principle "double negation is assertion" is formalized by the rule
\[ \neg\neg A \]
with the deduction rule
\[ \frac{\bot}{A} \text{⊥-C} \]
\[ \frac{\neg A, \Gamma \vdash \bot}{\Gamma \vdash A} \text{⊥-C} \]

Let \( \text{NK}^{\land \land} \) be the Natural Deduction system having the \( \bot \)-C rule in addition to the introduction and elimination rules for implication and conjunction.

(a) Write a derivation of \((A \rightarrow B) \rightarrow A\) in \( \text{NK}^{\land \land} \).

(b) Show: every derivation \( d \) in \( \text{NK}^{\land \land} \) can be transformed into a derivation \( d' \) where all conclusions \( A \) of a \( \bot \)-C rule are atomic.

2. The leftmost reduction strategy is the rule that prescribes to reduce always the leftmost redex, i.e., the redex \((\lambda x. u) t\) whose indicated parentheses occurs leftmost in the term.

The set \( \Lambda_I \) of the \( \lambda_I \) terms is defined as follows:

- every variable \( x \) belongs to \( \Lambda_I \);
- if \( u, t \in \Lambda_I \), then \((u)i \in \Lambda_I \);
- if \( u \in \Lambda_I \) and \( x \) actually occurs in \( u \), then \( \lambda x. u \in \Lambda_I \).

Show that every \( \lambda_I \)-term \( t \) is normalizable by leftmost reduction if and only if \( t \) is strongly normalizable.

Hint: Use induction on the lexicographical ordering of the pairs \((l(v), c(v))\), where \( l(v) \) is the length of the leftmost reduction of \( v \) and \( c(v) \) is the number of symbols of \( v \). Distinguish the cases

- \( v = \lambda x_1 \ldots \lambda x_m . ((\ldots (x) t_1 \ldots t_{n-1}) t_n) \);
- \( v = \lambda x_1 \ldots \lambda x_m . ((\ldots ((\lambda x. u)t) t_1 \ldots t_{n-1}) t_n) \), and notice that here \( x \) occurs in \( u \), hence if \( u[t/x] \) is strongly normalizable, then so is \( t \).