Assignment 1

Programming Theory

This assignment is worth 4 points in the final exam. For deadline and instructions for handing in, see the course homepage.
Proofs should be presented in exactly the same style as in the compendium (where there is no shortage of examples). It is not necessary to show the steps involving application of commutativity and associativity. You should, however, always state any arithmetic rule you use.

Problem 1 (25%) Prove the theorem

$$(\neg p \wedge (p \Rightarrow q)) = \neg p$$

using *only* the axioms and inference rules of Section 1 in the compendium.

Assignment 2 (25%) Prove

$$(p \wedge (p = q)) \Rightarrow ((q \Rightarrow r) = r)$$

using the axioms, theorems and inference rules of Section 1 and 2 in the compendium.

Assignment 3 (50%) Prove the theorem

$$\left(\begin{array}{c} (m_1 \leq m_2) \wedge (m_1 \leq m_3) \wedge (m_2 \leq m_3) \\ \wedge \\ (m_1 < m_2) \wedge (m_1 < m_3) \wedge (m_2 < m_3) \end{array}\right)$$
$$\Rightarrow$$
$$\left(\begin{array}{c} (\forall i : m_1 \leq i \leq m_2 : (b[i] > 0)) \wedge (\forall i : m_2 \leq i < m_3 : (b[i] > 0)) \\ = \\ \forall i : m_1 \leq i < m_3 : (b[i] > 0) \end{array}\right)$$

using the axioms, theorems and inference rules of Section 1, 2, 4, 5, 6 and 8 in the compendium.