10 Semantics of Programming Languages (nk480)

Regular expressions are defined by the following grammar:

$$
\begin{array}{rcll}
r & ::= & c & \text{Matches the single-character word } c \\
 & | & \epsilon & \text{Matches the empty word} \\
 & | & r_1 \circ r_2 & \text{Matches the concatenation of an } r_1\text{-word and an } r_2\text{-word} \\
 & | & 0 & \text{Matches no words} \\
 & | & r_1 + r_2 & \text{Matches any } r_1\text{-word or } r_2\text{-word} \\
 & | & r* & \text{Matches the concatenation of a finite number of } r\text{-words}
\end{array}
$$

(*a*) Give a set of inference rules defining a relation for when a word w is matched by a regular expression r. Use the notation $w \cdot w'$ to denote concatenation.

[8 marks]

(*b*) (*i*) Using the matching relation defined above, define a suitable notion of semantic equivalence $r_1 \simeq r_2$ for regular expressions. [4 marks]

(*ii*) Use this definition to prove that $(r + r') \simeq (r' + r)$. You may use inversion lemmas without proof, as long as they are explicitly indicated. [4 marks]

(*c*) Define an inductive relation $r\ \mathsf{null}$ characterizing the regular expressions r for which $\epsilon\ \mathsf{in}\ r$. [4 marks]