4 Prolog (ijl20)

In your answers ensure each relation which you define has a comment giving a declarative reading of its behaviour. Avoid unnecessary use of cut or other extra-logical relations. The library relations \(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_, \_\) atomic\(X\) may be used, the last succeeding if \(X\) is a number or atom e.g. \(42\) or \(abc\). Other relations should not be assumed.

(a) Define a relation \(\text{eval/2}\) to reduce arithmetic terms, so that e.g. \(\text{eval}(1+2*3,N)\) succeeds with \(N=7\). Atoms should reduce to themselves, so e.g. \(\text{eval}(a,Ans)\) succeeds with \(Ans=a\). [4 marks]

(b) Extend \(\text{eval}\) to allow function calls within arithmetic terms, such that \(2 * \text{apply}(\text{inc}, [2+3])\) reduces to \(12\). We will declare functions as Prolog facts in a \(\text{fun/2}\) relation, e.g. \(\text{inc}\) above will be specified by the fact \(\text{fun}(\text{apply}(\text{inc},[N]), N+1)\). Note the function arguments are held in a list, \([N]\) in this example, to support multi-argument functions. [6 marks]

(c) Extend relation \(\text{eval}\) to support a \(=\!/2\) operator, which reduces term \(A==B\) to \(\text{true}\) if \(A\) and \(B\) reduce to the same number or atom and \(\text{false}\) otherwise. For example \(\text{eval}(1+3==2+2,Ans)\) succeeds with \(Ans=true\). [3 marks]

(d) Extend relation \(\text{eval}\) to support terms of the form \(\text{if}(\text{Condition}, \text{Then}, \text{Else})\). These \(\text{if/3}\) terms should reduce to the reduction of either the \(\text{Then}\) term or the \(\text{Else}\) term determined by \(\text{Condition}\) reducing to \(\text{true}\) or \(\text{false}\). For example \(\text{eval}(\text{if}(1+2==4+5,a,b),Ans)\) succeeds with \(Ans=b\). Add a fact to the \(\text{fun}\) relation specifying the \(\text{factorial}\) function such that \(\text{eval}(\text{apply}(\text{fact},[5]),N)\) succeeds with \(N=120\). [7 marks]