

ToFoLo2 Project 3

Benjamin De Leeuw

Departement Zuivere Wiskunde en Computer Algebra
Galglaan 2, B-9000 Gent
Benjamin.DeLeeuw@UGent.be
Gent University, Belgium

December 14, 2005

1 Inleiding

Dit project gaat over Hoare Calculus en Prolog, zoals we beide hebben besproken in de oefeningenlessen. Er zijn slechts twee opdrachten: een Hoare Calculus bewijs, en een prolog predikaat.

2 Oplossing

De oplossing van *Oefening Hoare* bedraagt:

1. Een Fitch file met het Hoare Calculus bewijs, in hetzelfde formaat als in de oefeningenlessen van 07-12-2005: *Exercise Hoare.prf*.
2. Twee of meerdere bewijzen van de implicatieregels (deze werden in de oefeningen van 07-12-2005 met “...” aangegeven in de Fitch files). Gebruik LaTeX¹ of in ieder geval .pdf formaat voor deze wiskundige bewijzen. Groepeer al deze bewijzen in een file met naam *Exercise Hoare.pdf*. Leg ook duidelijk de link naar de Fitch file.

De oplossing van *Oefening Prolog* bedraagt een prolog file: *Exercise Prolog.pl*. Zip alle files in een bestand met als naam *GroepXX_P3.zip*, met *XX* uw groepsnummer op *Minerva*. Indien gebeurt via de *Dropbox* van *Minerva*.

3 Opgaven

3.1 Oefening Hoare

- Een specificatie van een functie $n!$:

$$1! = 1$$

$$k! = (k - 1)! \cdot k$$

Volgende programma implementeert deze functie, bewijs deze claim in Hoare Calculus:

¹Voor de installatie en het gebruik van LaTeX, verwijzen we naar de website <http://www.latex-project.org/>.

```

natural faculteit(natural n)
{
  natural fac := 1;
  natural count := n;
  while(count > 0)
  {
    fac := fac * count;
    count := count - 1;
  }
  return fac;
}

```

Zoek een invariant voor de while lus (inspiratie kan gevonden worden in de voorbeelden uit de oefeningenles van 07-12-2005). Bewijs dat de postconditie van het programma $\{fac = n!\}$ bedraagt, onder preconditionie $\{n > 0\}$. Opmerking: Het symbool “!” is gereserveerd in Fitch, gebruik dus het symbool “l” voor de faculteit (xl voor $x!$, $(x + 1)l$ voor $(x + 1)!$, enz.).

3.2 Oefening Prolog

Implementeer een predikaat *rotatie*/4 in prolog dat een lijst roteert naar links of naar rechts over een aantal posities. Voorbeelden met gewenste resultaat:

- ?-rotatie([1,2,3,4,5],links,2,[3,4,5,1,2]). → Yes.
- ?-rotatie([1,2,3,4,5],rechts,2,[4,5,1,2,3]). → Yes.
- ?-rotatie([1,2,3,4,5],links,2,L). → L = [3,4,5,1,2] No.

Maak eventueel gebruik van het **is ... mod** predikaat:

- mod5(X,Y) :- X **is** Y **mod** 5

Veel succes!