Exam Logical Verification

January 14, 2008

(6) exercises.
be given in Dutch or English. Good luck!

This exercise is concerned with first-order propositional logic
ıply typed λ-calculus (λ→).

oof in prop1 showing that the following formula is a tautology:

$$((A \to B \to A) \to A) \to A$$

type-derivation in λ→ corresponding to the proof in 1a.

the following simply typed λ-terms:

$$\lambda x :?.\, \lambda y :?.\, \lambda z :?.\, x\, z\, y$$
$$\lambda x :?.\, \lambda y :?.\, \lambda z :?.\, x\, (z\, y)$$
$$\lambda x :?.\, \lambda y :?.\, y\, x\, x$$

This exercise is concerned with first-order predicate logic (pred1)
with dependent types (λP).

ɔof in pred1 showing that the following formula is a tautology:

$$(\forall x.\, P(x) \to Q(x)) \to (\forall x.\, P(x)) \to (\forall x.\, Q(x))$$

λP-term corresponding to the formula in 2a.

ısed inhabitant in λP of the answer to 2b.


```
ıpresenting the propositions is declare
 prop : Set.
ıtion on prop is a binary operator *)
 imp : prop -> prop -> prop.
 use infix notation => for imp *)
" := imp (right associativity, at leve
esses if a proposion in prop is valid
p) is inhabited then p is valid
p) is not inhabited then p is not vali
r T : prop -> Prop.
```

ypes of the parameters imp_introduction and
troduction rule and elimination rule of the im

)

This exercise is concerned with Coq.

definition of an inductive datatype two with ex

s)

e induction principle for the datatype two.

ts)

er the following inductive definition of the pre

```
ive even : nat -> Prop :=
ı0 : even 0
ıSS : forall n:nat , even n -> even (S
```

if possible, an inhabitant of the following:

```
even 0
even 1
even 2
```

ɔints)

a fixed point definition of a function that ta
gives as output the sum of its elements (an
ty).

ɔints)

inal note is (the total amount of points plus 1