

	ANNEE UNIVERSITAIRE 2014 / 2015 SESSION D'AUTOMNE	Collège Sciences et technologies Masters
	MENTION : INFORMATIQUE PARCOURS / ETAPE : Master1 Code UE : J1IN7M21 Epreuve : Logique (Logic) Date : 15/12/2014 Heure : 8h30 Durée : 3h Documents : autorisés Epreuve de M : Castéran	

Any document is authorized
Recommended duration for this part: 1h30

Important: In questions 1, 2 and 3.2.2, you may answer either by giving proofs in natural deduction or a sequence of goals leading to the conclusion. **Any undocumented sequence of tactics will be ignored.**

1 Exercise

Prove the following propositions:

Lemma L0 : forall P Q R:Prop, (P → Q) → (P ∧ Q → R) → P → R.

Lemma L1: forall P:Prop, ~~P → (P ∨ ~P) → P.

2 Exercise

We consider the “principle of weak excluded middle” also known as *Jankov’s logic* obtained by adding to the intuitionistic logic of *Coq* the scheme $\neg A \vee \neg\neg A$ for any proposition A .

Definition weak_EXM := forall A:Prop, ~A ∨ ~~A

Prove that this principle implies the following proposition:

forall P Q : Prop, ~(P ∧ Q) → (~P ∨ ~Q)

3 Exercise

Let us recall the definition of binary trees:

```
Inductive bintree (A : Type) :=
| leaf
| node (label:A)(l_son r_son : bintree A).
```

The following predicate means “ There exists some node in t whose label is equal to a ”:

```

Fixpoint In_tree {A:Type} (a:A) (t : bintree A) :=
match t with
| leaf => False
| node r t1 t2 => a = r ∨ In_tree a t1 ∨ In_tree a t2
end.

```

3.1

Translate in *Coq* the following definitions:

A *total strict order* on a type A is a binary relation Lt on A which is

irreflexive: $Lt\ a\ a$ never holds

transitive: whenever $Lt\ a\ b$ and $Lt\ b\ c$, then $Lt\ a\ c$,

total: for any a and b , one of the three following propositions holds: $Lt\ a\ b$, $a = b$, or $Lt\ b\ a$

A binary tree of type `bintree A` is a *binary search tree* if

- The left subtree of any node contains only nodes with labels less than the node's label,
- The right subtree of any node contains only nodes with labels greater than the node's label,
- The left and right subtrees are also binary search trees.

Give in *Coq* a (polymorphic) definition of the predicate “being a binary search tree with respect to the relation Lt ”

3.2

We would like to prove that if Lt is a total strict order on A and $(\text{node } a\ t_1\ t_2)$ is a binary search tree, then no label can occur both in t_1 and t_2 .

3.2.1

Express this property as a proposition in *Coq*.

3.2.2

How do you prove it in *Coq*?