Examen - Ingénierie de la preuve

Durée : 3h00

Les notes de cours, de travaux dirigés et de travaux pratiques sont autorisées. Le sujet comporte 2 pages et les trois parties sont complètement indépendantes. Les règles logiques utiles pour ce contrôle sont redonnées à la fin du sujet. Le barème est donné à titre indicatif.

On s'attachera à soigner la présentation, en particulier lors des démonstrations par induction et on sera vigilant quant à la syntaxe lors de l'écriture de fragment de code devant être accepté par `Coq`*.*

1 Questions de cours (2 pts)

Question 1 On considère la définition inductive suivante :

```
Inductive X : nat -> Prop :=
| X0 : (X 1)
| X1 : forall n:nat, (X n) -> X (S (S n)).
```

Expliquer quel est le prédicat décrit par X. Expliquer le rôle de chacun des constructeurs X0 et X1.

Question 2 On considère la définition de la fonction puissance $x, n \mapsto x^n$ à savoir :

```
Fixpoint puissance (x:nat) (n:nat) {struct n} : nat :=
match n with
| O => 1
| S p => x * (puissance x p)
end.
```

Ecrire les règles de calcul associées à cette définition. On rappelle que les règles demandées sont celles qui s'appliquent lors de l'appel à la tactique `simpl`.

2 Logique et Isomorphisme de Curry-Howard (6 pts)

Question 3 Rappeler la différence entre logique intuitioniste et logique classique.
Question 4 Pour chacune des formules suivantes, proposer une démonstration sous forme d'arbre en déduction naturelle :

$$\neg(A \vee B) \rightarrow \neg A \wedge \neg B \qquad \neg A \wedge \neg B \rightarrow \neg(A \vee B).$$

On se limitera à l'application des règles de la logique intuitioniste.
Question 5 En déduire une suite de tactiques `Coq` prouvant $\neg A \wedge \neg B \rightarrow \neg(A \vee B)$.
Question 6 Déduire, toujours de la question 4, un terme de preuve démontrant l'énoncé suivant :

$$\neg A \wedge \neg B \rightarrow \neg(A \vee B).$$

Si nécessaire, on pourra utiliser les constructions suivantes ainsi que le fait que $\neg A$ est simplement un raccourci pour $A \rightarrow$ `False`.

```
or_ind : forall A B P : Prop, (A -> P) -> (B -> P) -> A \/ B -> P
and_ind : forall A B P : Prop, (A -> B -> P) -> A /\ B -> P
```

3 Définitions inductives et principe d'induction (12 pts)

On reprend l'exercice sur les pièces étudié lors du TP2.

Rappel de l'énoncé : *On souhaite démontrer qu'avec un nombre infini de pièces de 3 et 5 euros on peut faire l'appoint pour tout montant supérieur à 8 euros. Il s'agit donc de démontrer un théorème de la forme suivante :*

$$\forall m : \texttt{nat}, \exists i : \texttt{nat}, \exists j : \texttt{nat}, 8 + m = 5 * i + 3 * j.$$

Question 7 Rappelez le sens des variables m, i et j.

Question 8 On suppose que l'on dispose d'un principe d'induction particulier :

`mon_principe_d_induction` :
$\forall P : \texttt{nat} \to \texttt{Prop}, P\ 0 \to P\ 1 \to P\ 2 \to (\forall n : \texttt{nat}, P\ n \to P\ (n+3)) \to \forall n : \texttt{nat}, P\ n$

Expliquer comment vous démontreriez simplement le théorème en utilisant ce principe d'induction. Précisez les opérations à faire dans `Coq` pour y parvenir.
Question 9 Quels sont les trois cas de base et comment seront-ils prouvés ?
Question 10 Quel est le cas de récurrence et comment sera-t-il prouvé ?
Question 11 Le principe d'induction usuel sur les entiers naturels `nat_ind` est construit par filtrage et point fixe. On peut le définir en `Coq` de la manière suivante :

```
Fixpoint nat_ind (P:nat -> Prop)(H0 : P 0)(HR : forall n:nat, P n -> P (S n))(n:nat) : P n :=
  match n return (P n) with
    | 0 => H0
    | (S p) => HR p (nat_ind P H0 HR p)
  end.
```

Dans ce contexte, précisez quels sont les types des termes `H0` et `HR p (nat_ind P H0 HR p)` ? On notera que ces types sont différents, mais qu'ils sont unifiables en prenant en compte la forme de n donnée par le filtrage. Ainsi l'expression de filtrage est de type $P\ n$ comme indiqué par le mot-clé `return`.
Question 12 A partir de ce modèle, définissez en utilisant les notions de point-fixe et de filtrage, le principe d'induction `mon_principe_d_induction`.

A Rappel des règles logiques

On rappelle que $\bot \equiv \texttt{False}$ et $\neg A = A \to \bot$.

	règle d'élimination	règle(s) d'introduction
$\to \forall$	$\dfrac{\Gamma \vdash A \quad \Gamma \vdash A \to B}{\Gamma \vdash B}$	$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}$
$\wedge$	$\dfrac{\Gamma, A, B \vdash P \quad \Gamma \vdash A \wedge B}{\Gamma \vdash P}$	$\dfrac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$
$\vee$	$\dfrac{\Gamma, A \vdash P \quad \Gamma, B \vdash P \quad \Gamma \vdash A \vee B}{\Gamma \vdash P}$	$\dfrac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \quad \dfrac{\Gamma \vdash B}{\Gamma \vdash A \vee B}$
$\bot$	$\dfrac{\Gamma \vdash \bot}{\Gamma \vdash A}$	
$\neg$	$\dfrac{\Gamma \vdash A \quad \Gamma \vdash \neg A}{\Gamma \vdash \bot}$	$\dfrac{\Gamma, A \vdash \bot}{\Gamma \vdash \neg A}$
$\exists$	$\dfrac{\Gamma, x : A, P\ x \vdash Q \quad \Gamma \vdash \exists x : A, P\ x}{\Gamma \vdash Q}$	$\dfrac{\Gamma, v : A \vdash P\ v}{\Gamma \vdash \exists x : A, P\ x}$