L'assistant à la preuve Coq,
Le calcul des prédicats

Polytech Nice-Sophia
Département informatique
Preuves de programmes
Sylvain Lippi

Exercice 1 (pour tout) *Déclarer les variables E, P et Q (avec les types qui conviennent) et prouver les théorèmes suivants :*

$$(\forall x : E, P\ x) \vee (\forall y : E, Q\ y) \to (\forall z : E, P\ z \vee Q\ z)$$
$$(\forall x : E, P\ x \to Q\ x) \to (\forall y : E, P\ y) \to \forall z : E, Q\ z$$

Remarques syntaxiques : ne pas oublier l'espace entre un prédicat et sa variable. Les quantificateurs $\forall$ et $\exists$ s'écrivent respectivement `forall` *et* `exists`*.*

Exercice 2 (il existe) *Déclarer les variables E, P et Q (avec les types qui conviennent) et prouver les théorèmes suivants :*

$$(\exists x : E, P\ x \vee Q\ x) \to (\exists y : E, P\ y) \vee (\exists z : E, Q\ z)$$
$$(\exists x : E, P\ x) \vee (\exists y : E, Q\ y) \to (\exists x : E, P\ x \vee Q\ x)$$
$$(\forall x : E, P\ x) \to \neg(\exists y : E, \neg P\ y)$$
$$(\exists a : E, P\ a) \to (\forall x : E, P\ x \to Q\ x) \to \exists a : E, Q\ a$$
$$\neg(\exists x : E, \forall R : E \to Prop, R\ x)$$

Indication. Pour prouver le dernier théorème, on pourra définir et utiliser le prédicat qui vaut `False` pour tous les éléments $x \in E$.

Exercice 3 (Des parenthèses... et de l'intuitionisme) *Déclarer les variables E et P (avec les types qui conviennent) et* essayer *de prouver les théorèmes suivants :*

$$\forall x : E, P\ x \to \exists x : E, P\ x$$
$$(\forall x : E, P\ x) \to \exists x : E, P\ x$$

Exercice 4 (peirce et ses amis) *On propose différentes caractérisations de la logique classique. Définir les propositions suivantes et montrer qu'elles sont équivalentes.*

`peirce`	$\forall P\ Q : Prop,\ ((P \to Q) \to P) \to P$
`nono`	$\forall P : Prop,\ \neg\neg P \to P$
`tiers`	$\forall P : Prop,\ P \vee \neg P$
`demorgan`	$\forall P\ Q : Prop,\ \neg(\neg P \wedge \neg Q) \to P \vee Q$
`flèche`	$\forall P\ Q : Prop,\ (P \to Q) \to (\neg P \vee Q)$

Indication. On pourra, par exemple, prouver les cinq théorèmes suivants : `peirce` $\to$ `nono` *(appliquer* `peirce` *avec $Q := \neg P$),* `nono` $\to$ `tiers`, `tiers` $\to$ `demorgan` *(détruire* `tiers` *avec $P := P \vee Q$),* `demorgan` $\to$ `flèche` *et* `flèche` $\to$ `peirce` *(détruire* `flèche` *avec $Q := P$)*

Exercice 5 (Klüb des buveurs) *On se propose de formaliser le paradoxe des buveurs, dû à Smullyan : «*Dans toute pièce non vide on peut trouver une personne ayant la propriété suivante : si cette personne boit, alors tout le monde dans la pièce boit.*»*

Déclarer avec la commande `Axiom` *l'axiome du tiers-exclu, formaliser l'énoncé précédent (attention aux parenthèses :* $\exists$ *a priorité sur* $\rightarrow$*) et en faire la preuve.*

Indication. *Le tiers-exclu permet de faire des preuves par cas avec la tactique* `destruct`*. En particulier, on considèrera deux cas : il existe quelqu'un qui ne boit pas... ou pas ! De plus, on pourra utiliser le tiers-exclu plusieurs fois au cours de la preuve ; les hypothèses ne sont pas à usage unique...*

Exercice 6 (Klüb écossais) *Voici les règles adoptées par un club très privé :*

1. *Tout membre qui n'est pas écossais porte des chaussettes rouges.*
2. *Tout membre porte un kilt ou ne porte pas des chaussettes rouges.*
3. *Les membres mariés ne sortent pas le dimanche.*
4. *Un membre sort le dimanche si et seulement s'il est écossais.*
5. *Tout membre qui porte un kilt est écossais et marié.*
6. *Tout membre écossais porte un kilt.*

En utilisant la logique classique, montrer que personne ne peut être accepté par ce club.