Práctico 1

Asistentes de pruebas para lógicos y matemáticos

Cálculo Proposicional

Objetivos

Adquirir el manejo básico de conectivos y de tácticas. Familiarizarse con el ambiente de pruebas Coq.

Principales tácticas a utilizar en estos ejercicios

`intro:` introduce la primera hipótesis del objetivo corriente en el contexto de hipótesis.

`intros:` introduce todas las hipótesis del objetivo corriente en el contexto de hipótesis.

`intros <nom_1>...<nom_n>`**:** introduce las n primeras hipótesis del objetivo corriente, en el contexto de hipótesis nombrándolas `<nom_1>...<nom_n>`.

`apply <nom>:` para probar el objetivo corriente aplicando el lema o la hipótesis `<nom>`. Corresponde a aplicar la regla de Modus Ponens de abajo hacia arriba.

`assumption:` usa una hipótesis.

`split:` para probar un $\wedge$ probando cada uno de sus componentes (introducción del $\wedge$).

`left:` para probar un $\vee$ probando el primero de sus componentes (introducción del $\vee$ a izquierda).

`right:` para probar un $\vee$ probando el segundo de sus componentes (introducción del $\vee$ a derecha).

`absurd <prop>`**:** para probar el objetivo Falso probando `<prop>` y `~<prop>` .

`elim <nom>`**:** elimina la conectiva principal de la hipótesis o el lema `<nom>`.

`unfold <nom> [in h]:` expande la definición de `<nom>` en el objetivo corriente (o en la hipótesis h).

`exact <nom>:` para probar el objetivo explicitando la prueba `<nom>`. Puede ser una hipótesis o un lema definido previamente en el contexto.

`cut <prop>:` para probar el objetivo corriente probando el lema intermedio `<prop>`**.**

<u>Nota</u>: Algunas de las tácticas previas que toman argumentos requieren que éstos sean escritos entre paréntesis. En particular cuando se trata de fórmulas compuestas de tipo `Prop`.

Lógica Minimal

Usa `intro(s), apply, assumption` (entre otros).

Ejercicio 1.1. Demuestre en Coq los siguientes teoremas *I*, *K* y *S*.

1. $A \to A$
2. $A \to B \to A$
3. $(A \to (B \to C)) \to (A \to B) \to (A \to C)$

Ejercicio 1.2. Demuestre en Coq los siguientes teoremas.

1. $(A \to B) \to (B \to C) \to A \to C$
2. $(A \to B \to C) \to B \to A \to C$

Ejercicio 1.3. Construya dos pruebas diferentes para los siguientes teoremas:

1. $A \to A \to A$
2. $(A \to B \to C) \to A \to (A \to C) \to B \to C$.

Cálculo Proposicional Constructivo

Usa `split`, `left`, `right`, `absurd`, `elim`, `unfold` (entre otros).
Recuerde que ~A es una abreviación de $A \to False$, y $A \leftrightarrow B$ es una abreviatura de $(A \to B) \wedge (B \to A)$.

Ejercicio 1.4. Demuestre en Coq los siguientes teoremas:

1. $A \to$ ~~A. (Sugerencia: usar `unfold not`)
2. $A \to B \to A \wedge B$
3. $(A \to B \to C) \to (A \wedge B \to C)$
4. $A \to (A \vee B)$
5. $B \to (A \vee B)$
6. $(A \vee B) \to (B \vee A)$
7. $(A \to C) \to (B \to C) \to A \vee B \to C$
8. $False \to A$

Ejercicio 1.5. Demuestre en Coq los siguientes teoremas:

1. $(A\to B)\to \sim B\to \sim A$
2. $\sim(A\wedge\sim A)$
3. $(A\to B)\to \sim(A\wedge\sim B)$
4. $(A\wedge B)\to\sim(A\to\sim B)$
5. $\sim A\wedge\sim\sim A\to$False.

Ejercicio 1.6. Demuestre en Coq los siguientes teoremas:

1. $A\vee B\to\sim(\sim A\wedge\sim B)$
2. $A\vee B\leftrightarrow B\vee A$
3. $A\vee B\to((A\to B)\to B)$

Cálculo Proposicional Clásico

Usa `cut`, `exact` (entre otros).

Las tácticas definidas hasta el momento sólo nos permiten probar teoremas de la Lógica Constructiva. Para poder trabajar en lógica Clásica, debemos agregar algún axioma que permita derivar resultados clásicos. El axioma clásico por excelencia es la fórmula $A\vee \sim A$.

Ejercicio 1.7. Demuestre en Coq los siguientes teoremas de la lógica clásica:

1. $A\vee \sim A \to \sim\sim A\to A$
2. $A\vee \sim A \to(A\to B)\vee(B \to A)$
3. $(A\vee \sim A) \to \sim(A\wedge B)\to \sim A \vee \sim B$ (de Morgan $\to$)

Notas

Observe que en todos los lemas anteriores, debimos agregar como antecedente a la fórmula $A\vee \sim A$. Para poder utilizar esta fórmula como un axioma y no tener que explicitarla en cada lema, escriba la siguiente secuencia de comandos en Coq:

```
Reset Initial. (* no es aceptado el Reset en la IDE de Coq *)
Require Import Classical.
Check classic.
```

El primero de estos comandos borra del contexto de definiciones de la sesión. El segundo comando carga el archivo `Classical`, que es el que se usa para trabajar con Lógica Clásica, y el tercero informa el tipo del axioma `classic`, que se encuentra definido en el archivo

`Classical`. Este axioma es de tipo (forall P: Prop, P∨ ~P). Esto debe leerse del siguiente modo: *Para toda proposición P, se cumple P ∨ ~P*.

Observe que todos los lemas probados hasta ahora, deben considerarse como *esquemas* de lemas, ya que se cumplen para cualquier instanciación de las variables A, B y C. La forma de indicar en Coq que una fórmula es un esquema es justamente cuantificando universalmente a las variables proposicionales que aparecen en ella.

En general, (forall x:A, B) se lee como *Para todo x en A, se cumple B*, y x puede aparecer en B.

Ejercicio 1.8. Utilizando ahora el axioma `classic` demuestre la siguiente reformulación de los ejercicios del punto 1.7. Para aplicar el axioma `classic` debe instanciarlo explícitamente en la fórmula que necesite. Por ejemplo para instanciar `classic` con la proposición ~A escriba: `(classic ~A)`. Por ejemplo: `(classic False): False ∨ ~False`.

1. forall A: Prop, ~~A→A
2. forall A B: Prop, (A→B)∨(B →A)
3. forall A B: Prop, ~(A∧B)→ ~A ∨ ~B

Razonamiento Automatizado

Ejercicio 1.9. Demuestre algunos ejercicios de las dos primeras secciones utilizando la táctica automática `tauto`. Observe que los teoremas de la lógica clásica no se pueden demostrar con esta táctica.

Ejemplos de Traducción al Lenguaje Lógico

Ejercicio 1.10. Considere el siguiente problema: Un club escocés tiene las siguientes reglas para aceptar a sus miembros:

Regla 1: Todo no escocés debe usar medias rojas.

Regla 2: Todo miembro usa kilt o no usa medias rojas.

Regla 3: Los miembros casados no salen los domingos.

Regla 4: Un miembro sale los domingos si y sólo si es escocés.

Regla 5: Todo miembro que usa kilt es escocés y es casado.

Regla 6: Todo miembro escocés usa kilt.

1. Formalice el problema en Coq.
2. Demuestre que las reglas son contradictorias (o sea que nadie puede ser aceptado).

3. Pruebe que las reglas son contradictorias usando la táctica `tauto`.

Ejercicio 1.11. Considere las siguientes reglas extraídas de un manual de patología:

Regla 1: Si el paciente tiene fiebre o el paciente tiene la piel amarillenta, entonces tiene hepatitis o rubeola.

Regla 2: Si el paciente tiene rubeola, entonces tiene fiebre.

Regla 3: Si el paciente tiene hepatitis, pero no la rubeola, entonces tiene la piel amarillenta.

En el momento del examen clínico, usted observa los siguientes hechos:

- El paciente no tiene la piel amarillenta.
- El paciente tiene fiebre.

Establezca el diagnóstico y demuestre en Coq que su diagnóstico es correcto.

Ejercicios a entregar: 1.3, 1.4, 1.6, 1.8 y 1.11.
Ver fecha de envío en el calendario de entregas, en el sitio web del curso.

El archivo a entregar (NombreApellido.v) debe compilar correctamente en Coq. Usar la plantilla publicada junto con el práctico para el desarrollo del mismo.