



Universidad de Salamanca

***Tableaux* semánticos**

&

Tableau III

Asignatura: Lógicas para la Informática y la Inteligencia Artificial

Alumna: Almudena Vicente Tocino

Tabla de contenidos

1	Introducción	3
2	<i>Tableaux</i> semánticos.....	4
3	Tableau III	6
3.1	Ejemplo de uso.....	8
3.2	Marcar () las sentencias en los tableaux de predicados.....	14
3.3	Funcionalidad "User input"	15
3.4	Ayuda	15
4	Conclusiones.....	16
5	Anexo I: Requisitos del sistema	17
6	Bibliografía	18

Tabla de figuras

Ilustración 1. Tableau III. 25 Ejercicios lógica proposicional disponibles.....	6
Ilustración 2. Tableau III. 32 Ejercicios lógica de predicados disponibles.....	7
Ilustración 3. Tableau III. 24 Ejercicios lógica de predicados con identidad disponibles.....	8
Ilustración 4. Tableau III. Seleccionar tipo de ejercicio	9
Ilustración 5. Tableau III. Seleccionar ejercicio	9
Ilustración 6. Tableau III. Ejemplo de aplicación de reglas.....	10
Ilustración 7. Tableau III. Ejemplo de aplicación de reglas.....	11
Ilustración 8. <i>Tableu</i> resuelto	11
Ilustración 9. Tableau III. Funcionalidad " <i>Tell me which rule to use</i> "	12
Ilustración 10. Tableau III. Funcionalidad " <i>Apply a rule for me</i> "	13
Ilustración 11. Tableau III. Ejercicio finalizado	14

1 Introducción

El objetivo de este documento es presentar el programa **Tableau III**. Este software ha sido desarrollado por la Universidad de Oxford (Reino Unido). En esta universidad, existe un grupo de trabajo que mantiene un sitio Web con material y herramientas que sirven como introducción al estudio de la lógica elemental, cubriendo el cálculo proposicional y de predicados.

Entre el material disponible, se encuentra el software mencionado, **Tableau III**, que fue creado con la intención de permitir a los alumnos practicar haciendo *tableaux* para la lógica proposicional y de predicados. **Tableau III** es similar en funcionalidad a otro programa creado dentro del mismo grupo de trabajo, **Tableau2**, pero escrito en Java (en vez de DOS).

Antes de comentar en detalle la funcionalidad que **Tableau III** ofrece, se consideró oportuno incluir una breve introducción teórica al concepto de *tableaux*. Por tanto, la siguiente sección intentará cubrir los aspectos más relevantes de la definición, construcción y reglas de los *tableaux*. Posteriormente, se detallarán los aspectos más significativos de **Tableau III** junto con algunos ejemplos de uso. Por último, incluyen unas breves conclusiones del trabajo expuesto.

Cabe señalar que la documentación teórica que se incluye en este trabajo ha sido extraída tanto de la información que se ha visto durante la asignatura, así como dos artículos de Ángel Nepomuceno que tratan en amplitud el tema de los *tableaux* semánticos. En el apartado de bibliografía, dichos artículos son referenciados.

2 Tableaux semánticos

Las tablas o árboles semánticos son un mecanismo de decisión para sentencias de un lenguaje formal.

Como reseña histórica, el procedimiento de árboles semánticos se le atribuye al autor Beth y su "*Semantic entailment and formal derivability*", donde aparece el término de tabla semántica. Beth la define como un intento de búsqueda sistemática de contraejemplos para determinar si una fórmula es consecuencia de otras.

La siguiente definición nos ayudará a comprender el concepto de árbol semántico, así como el proceso de construcción: "*Un árbol semántico es una sucesión de sucesiones de fórmulas llamadas ramas, generadas a partir de un conjunto (no vacío) de fórmulas, por aplicación a éstas de las reglas (y a las fórmulas resultantes que sean complejas).*"¹

El proceso de construcción de una rama finaliza cuando aparecen dos literales contradictorios o cuando todas las fórmulas que no son literales están marcadas. Una rama está cerrada cuando en ella aparece un literal y su contradicción. Una rama está acabada si está cerrada o todas sus sentencias no literales están marcadas.

Para la construcción de *tableaux* semánticos tenemos que tener en cuenta los siguientes tipos de fórmulas:

- Fórmula- α : Cada parte de una conjunción
- Fórmula- β : Cada parte de una disyunción. Además de la negación del antecedente y el consecuente de una implicación
- Fórmula- γ : Instancias de la matriz de una fórmula cuyo signo principal es $\forall$
- Fórmula- δ : Instancias de la matriz de una fórmula cuyo signo principal es $\exists$

Por último, este resumen de *tableaux* semánticos incluye las reglas que se pueden aplicar a cada una de las fórmulas que aparezcan en un *tableau*:

1. Sea $\neg \alpha$ la fórmula a marcar

1. si α es $\neg \beta$, anotar β

¹ "Enseñar lógica: lenguaje lógico y árboles semánticos" Ángel Nepomuceno

2. si α es $\beta \wedge \gamma$, anotar $\neg\beta \vee \neg\gamma$
 3. si α es $\beta \vee \gamma$, anotar $\neg\beta \wedge \neg\gamma$
 4. si α es $\beta \rightarrow \gamma$, anotar $\beta \wedge \neg\gamma$
 5. si α es $\exists x\beta$, anotar $\forall x\neg\beta$
 6. si α es $\forall x\beta$, anotar $\exists x\neg\beta$
2. si la fórmula a marcar es $\beta \wedge \gamma \rightarrow$ anotar consecutivamente β y γ al término de la rama y marcar con $\wedge$
 3. si la fórmula a marcar es $\beta \vee \gamma \rightarrow$ abrir dos subramas, una con β y otra con γ y marcar con $\vee$
 4. si la fórmula a marcar es $\beta \rightarrow \gamma \rightarrow$ abrir dos subramas, una con una con $\neg\beta$ y otra con γ y marcar con $\rightarrow$
 5. si la fórmula a marcar es $\exists x\beta \rightarrow$ anotar $\beta(b_k/x)$ al término de la rama, b_k es el parámetro de menor subíndice que no ocurriría en ninguna fórmula de la rama, se marca la fórmula con dicho parámetro
 6. si la fórmula a marcar es $\forall x\beta \rightarrow$ anotar $\beta(b_1/x), \dots, \beta(b_n/x)$ al término de la rama, $b_1, \dots, b_n$ todos los parámetros que ocurren en la rama,; a medida que van anotándose $\beta(b_i/x)$, se marca la fórmula con b_i

Un último inciso, en un determinado momento en la construcción de un *tableau* podemos disponer de varias fórmulas sin marcar, por lo que podemos aplicar las reglas a una fórmula u otra. No hay una norma para decidir qué fórmula marcar. Existe una estrategia, se da prioridad a las fórmulas según su signo lógico principal. Dispondremos de dos bloques de precedencia:

- Sentencias proposicionales
- Sentencias cuantificacionales

se suele aconsejar considerar como prioritarias las sentencias proposicionales. Entonces, el orden de prioridad será $\neg, \wedge, \vee / \rightarrow, \exists, \forall$

3 Tableau III

Tableau III fue creado con la intención de permitir a los alumnos practicar haciendo *tableaux* para la lógica proposicional y de predicados. **Tableau III** es similar en funcionalidad a otro programa creado dentro del mismo grupo de trabajo, **Tableau2**, pero escrito en Java (en vez de DOS).

En la aplicación podemos seleccionar del menú *Exercises* distintos tipos de ejercicios:

- Lógica proposicional: permite el razonamiento, a través de un mecanismo que primero evalúa sentencias simples y luego sentencias complejas, formadas mediante el uso de conectivos proposicionales, por ejemplo Y (AND), O (OR). Para resolver estos ejercicios se tienen que hacer uso de las reglas 1, 2, 3, 4 expuestas en el capítulo anterior.

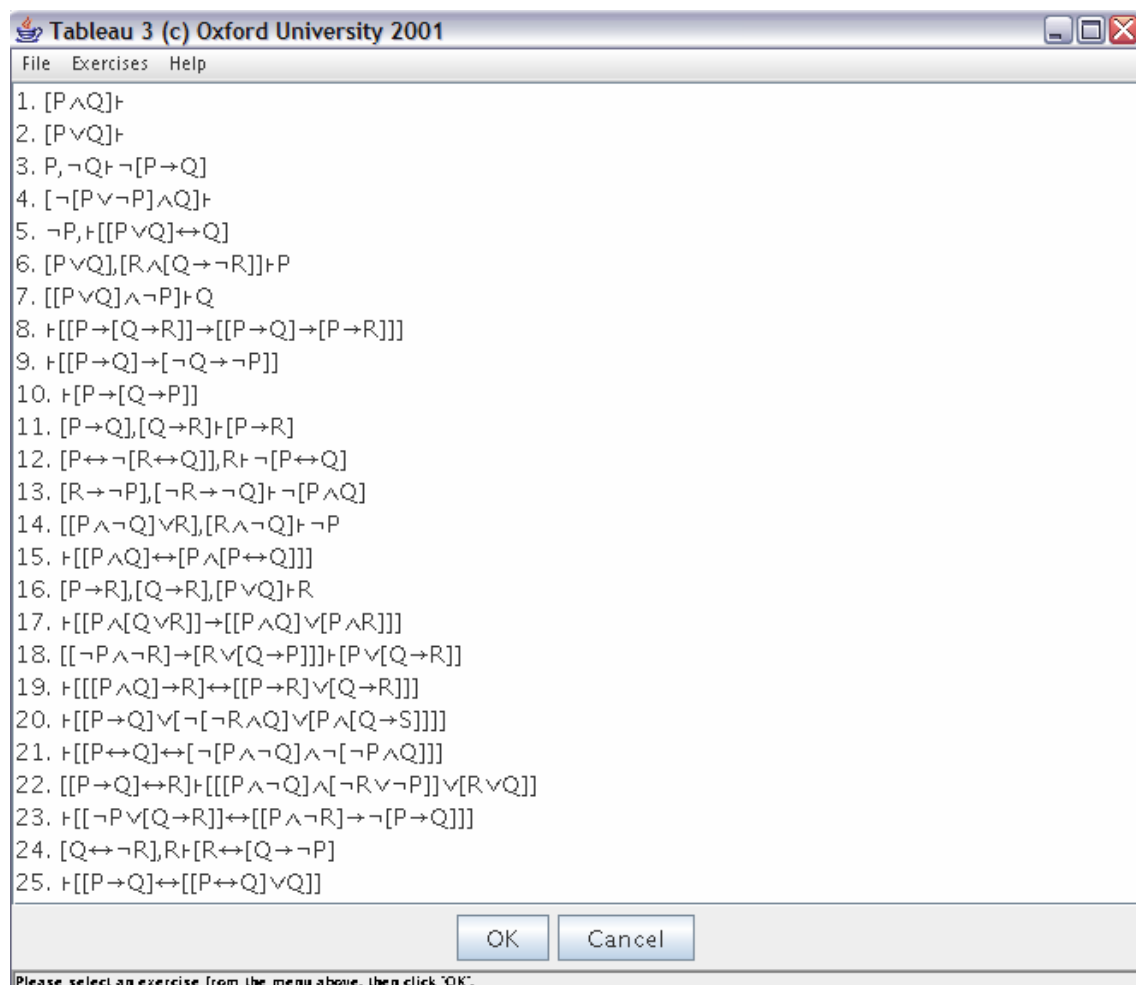


Ilustración 1. Tableau III. 25 Ejercicios lógica proposicional disponibles

- Lógica de predicados: añade a la lógica proposicional el concepto de cuantificador universal y existencial. Para resolver estos *tableaux* se hará uso de las 6 reglas expuestas en el capítulo anterior.

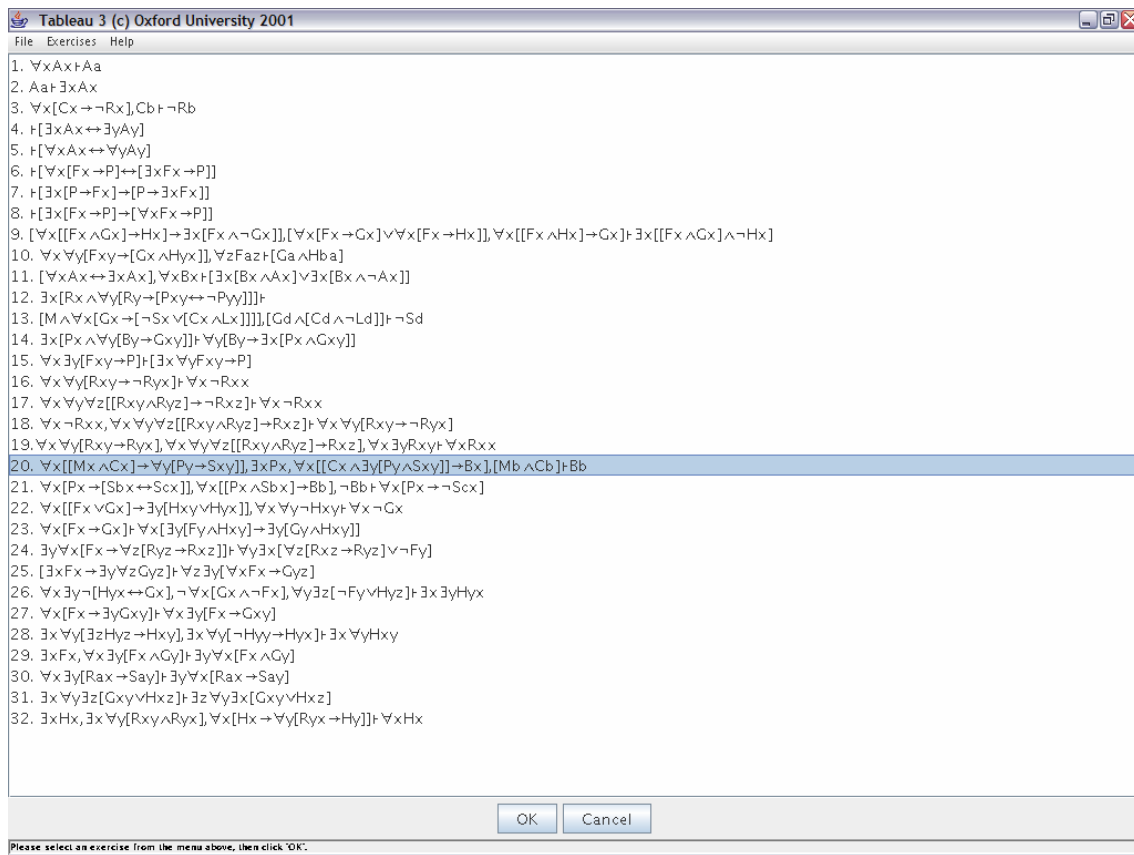


Ilustración 2. Tableau III. 32 Ejercicios lógica de predicados disponibles

- Lógica de predicados con identidad: con la lógica proposicional y de predicados definidas y dado este predicado:

$\exists x, F(x)$ siendo $F(x) = x$ es una flor

sólo podemos asegurar que hay al menos una flor. Para decir que hay al menos dos flores necesitamos hacer uso del concepto de identidad. Se usa " $x=y$ " para decir que x es la misma cosa que y . Con este formalismo ahora podemos introducir predicados del tipo:

$\exists x, \exists y[[F(x) \wedge F(y)] \wedge \neg x=y] \rightarrow$ Hay al menos dos flores (x, y)

Para este tipo de fórmulas necesitamos dos reglas adicionales para la construcción de *tableaux*.

La regla de la izquierda ($D=E$) y la regla de la derecha ($E=D$)

$$\frac{\Phi}{E=D} \frac{}{\Psi}$$

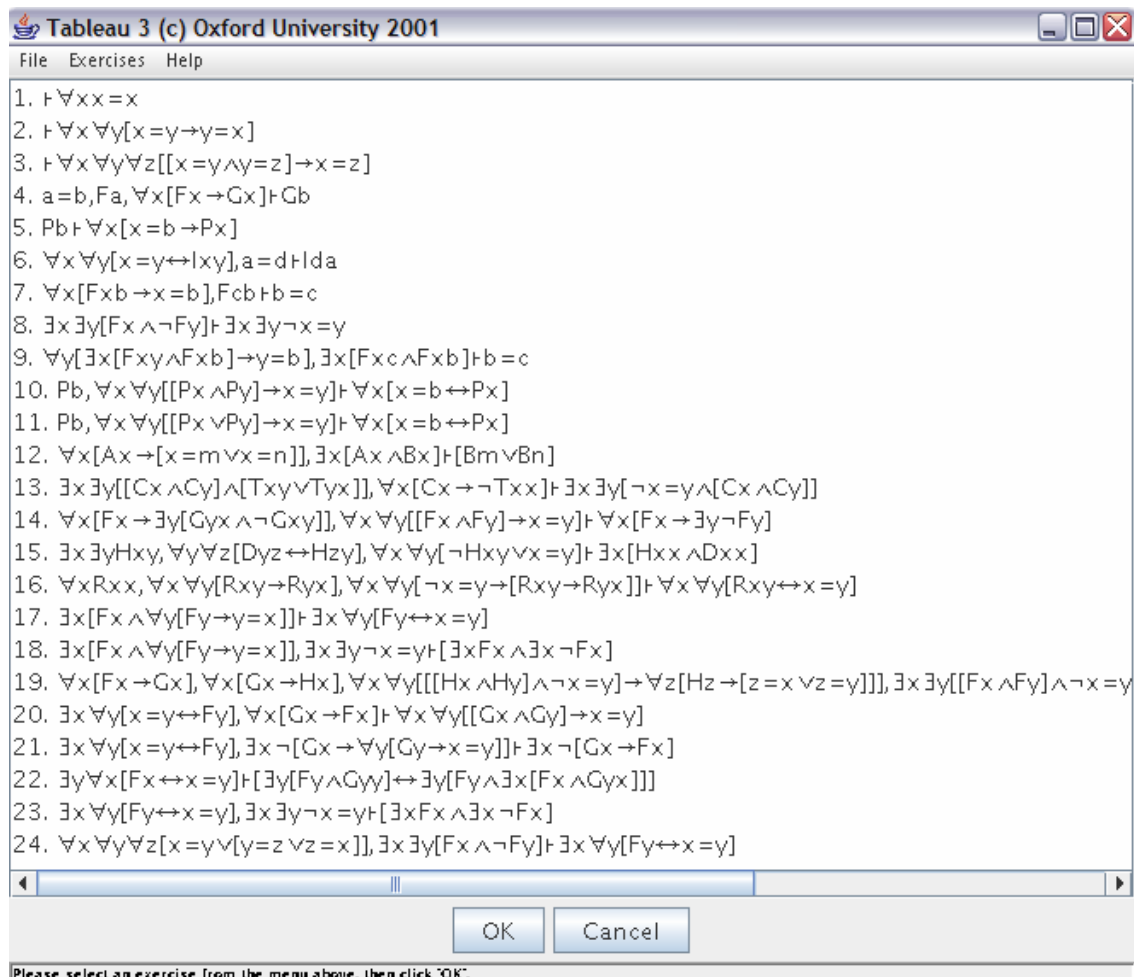
$$\frac{\Phi}{D=E} \frac{}{\Psi}$$


Ilustración 3. Tableau III. 24 Ejercicios lógica de predicados con identidad disponibles

3.1 Ejemplo de uso

Esta subsección mostrará los pasos que se deben dar para que los usuarios puedan practicar con el software **Tableau III**. Como ejemplo, se realizará un ejercicio de lógica de predicados. Se ha considerado útil incluir capturas de pantallas para facilitar la comprensión de cada uno de los pasos que se explican.

Primero, el usuario tiene que seleccionar del menú *Exercises* y a continuación el submenú *Predicate*, como muestra la Ilustración 4

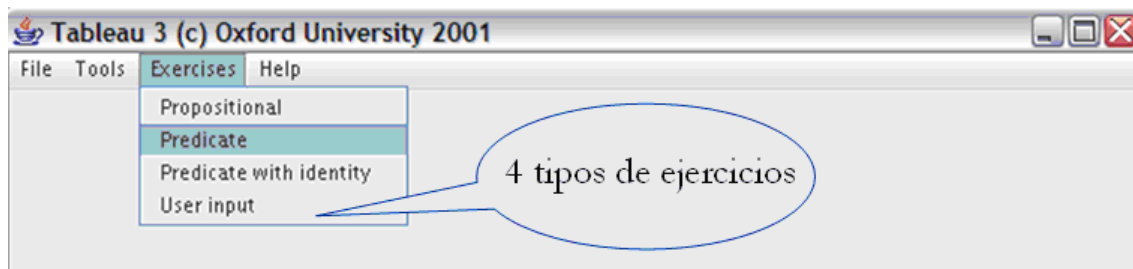


Ilustración 4. Tableau III. Seleccionar tipo de ejercicio

Una vez seleccionado el tipo de ejercicio, lógica de predicados en nuestro caso, se presenta un listado con todos los posibles ejercicios que el usuario puede desarrollar, como se puede ver en la Ilustración 5.

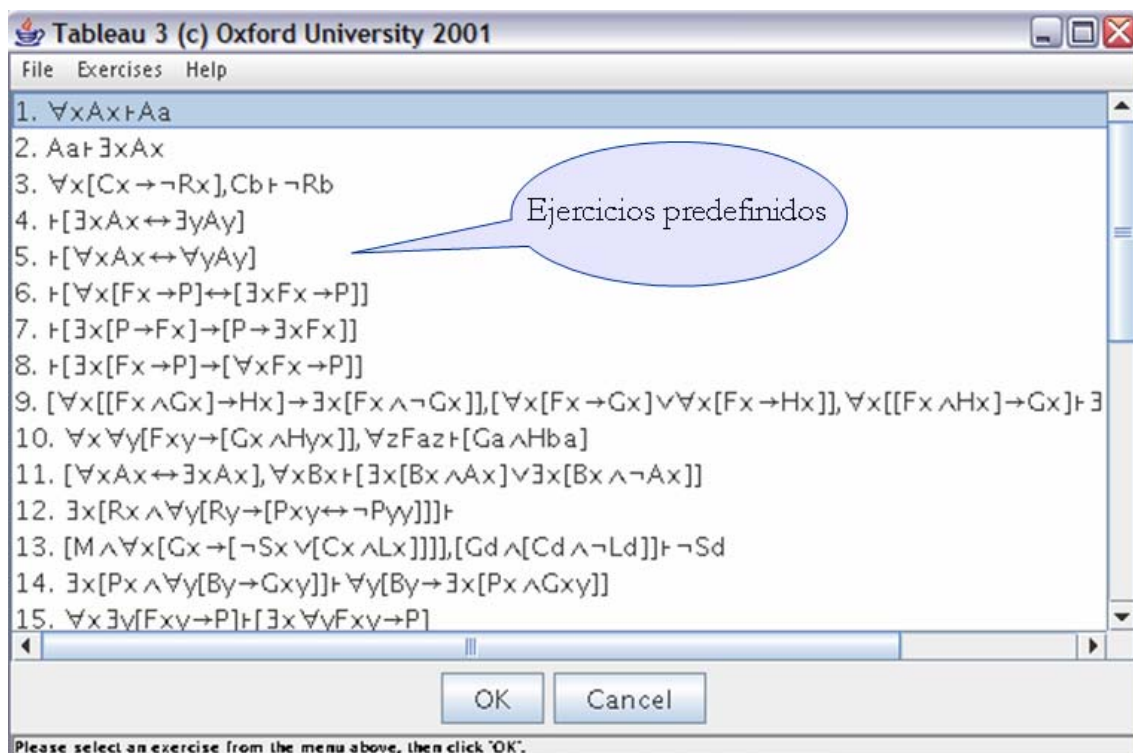


Ilustración 5. Tableau III. Seleccionar ejercicio

Una vez que el usuario ha seleccionado el ejercicio, en este ejemplo se trabaja con este ejercicio $\forall x[Cx \rightarrow \neg Rx], Cb \vdash \neg Rb$, se le muestra en pantalla la fórmula que tendrá que desarrollar. El funcionamiento es simple, el usuario sólo tiene que seleccionar con el botón derecho del ratón una sentencia no marcada del *tableau* y se desplegará un listado con todas las reglas. La Ilustración 6 muestra esta situación, el usuario ha seleccionado la sentencia $\forall x[Cx \rightarrow \neg Rx]$ y le indica al programa la regla que quiere aplicar. En este caso la regla correcta es $\forall x\varphi$.

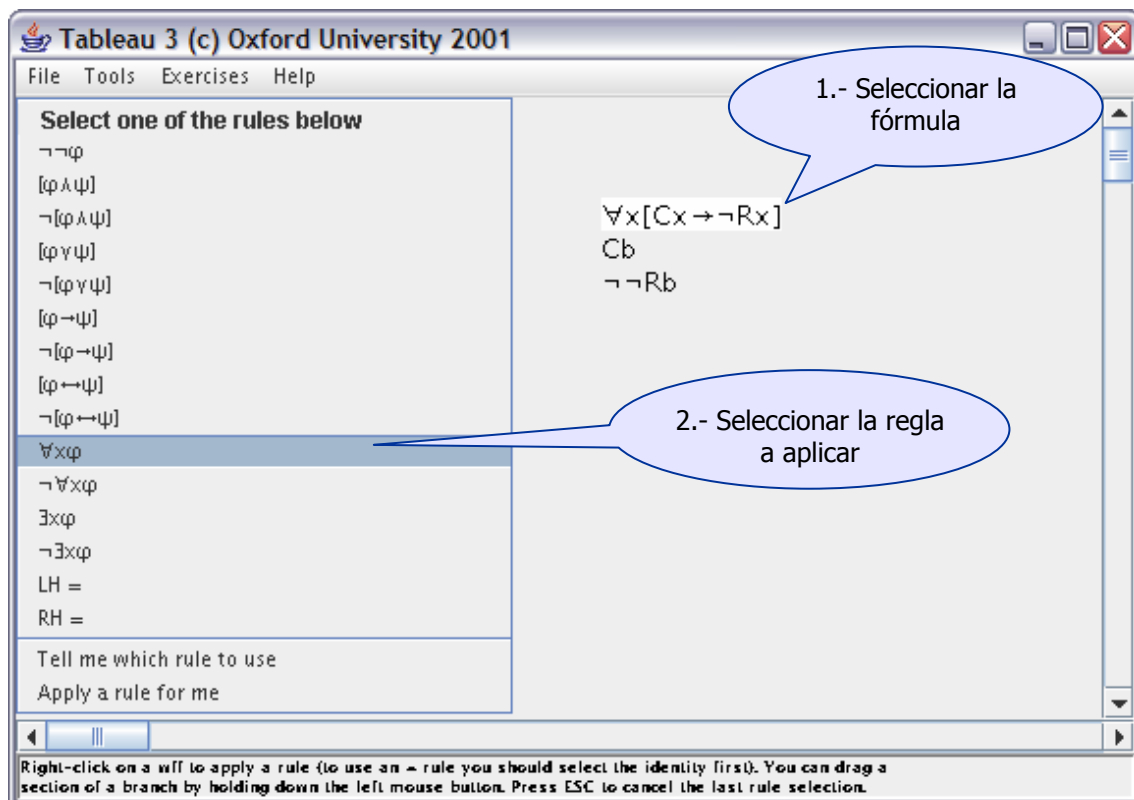


Ilustración 6. Tableau III. Ejemplo de aplicación de reglas

Cuando el usuario selecciona la regla correcta, la aplicación muestra un campo de texto donde el usuario tiene que introducir la fórmula resultante de aplicar la regla $\forall x\phi$ a la sentencia seleccionada, $\forall x[Cx \rightarrow \neg Rx]$. En este caso la fórmula resultante es $[Cb \rightarrow \neg Rx]$. Si el usuario introduce cualquier otra fórmula, la aplicación muestra mensajes de error en la barra de estado indicando que la solución no es adecuada. La Ilustración 7 muestra esta situación.

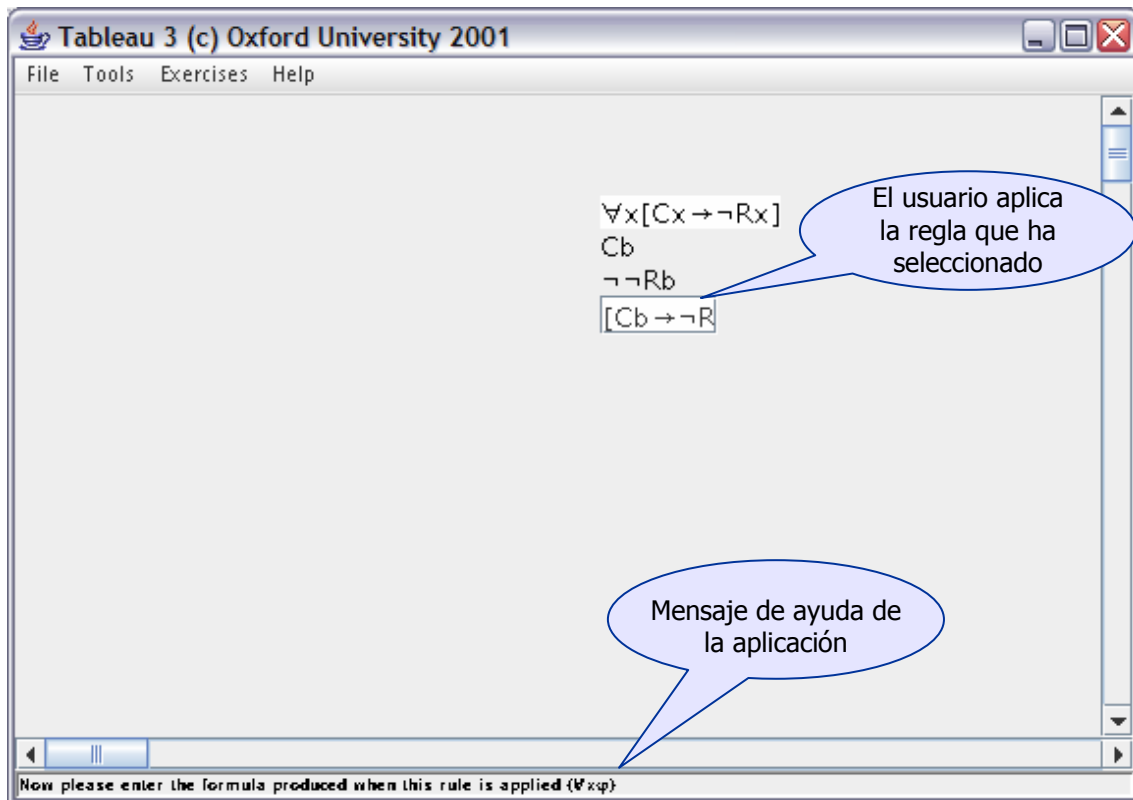


Ilustración 7. Tableau III. Ejemplo de aplicación de reglas

Repitiendo este proceso de selección de fórmula, selección de regla y aplicación de la regla seleccionada a la fórmula con la que está trabajando, el usuario puede resolver los *tableaux*, hasta que estos estén cerrados o no se le puedan aplicar más reglas.

El *tableau* resultante del ejercicio $\forall x[Cx \rightarrow \neg Rx], Cb \vdash \neg Rb$ que se estaba planteando como ejemplo es:

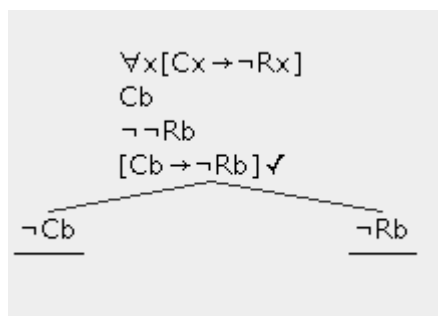


Ilustración 8. Tableau resuelto

En el caso de que el usuario esté confuso y no sepa que regla aplicar, el programa ofrece dos funcionalidades para ayudarle en la resolución de un *tableau*. La Ilustración 10 e Ilustración 9 muestran estas opciones.

- *Tell me which rule to use*: muestra en la barra de estado de la ventana principal de la aplicación un mensaje de ayuda con la regla que el usuario debe aplicar, pero deja a éste la responsabilidad de aplicar la regla

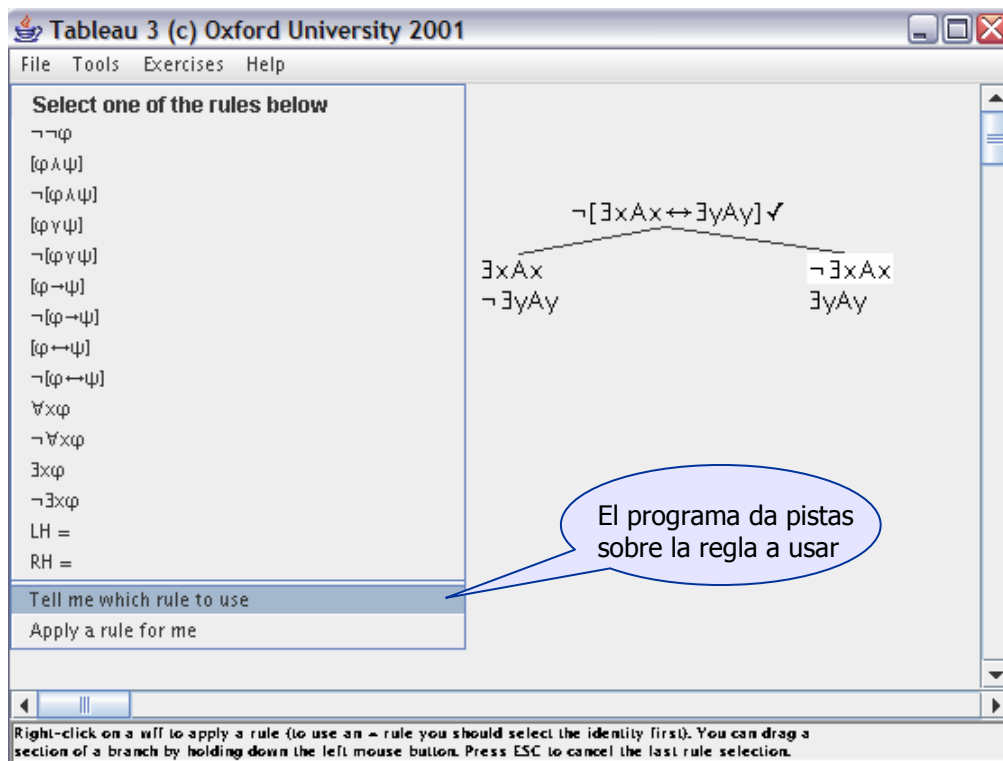


Ilustración 9. Tableau III. Funcionalidad "*Tell me which rule to use*"

- *Apply rule for me*: el programa selecciona y aplica la regla correspondiente según la sentencia seleccionada por el usuario.

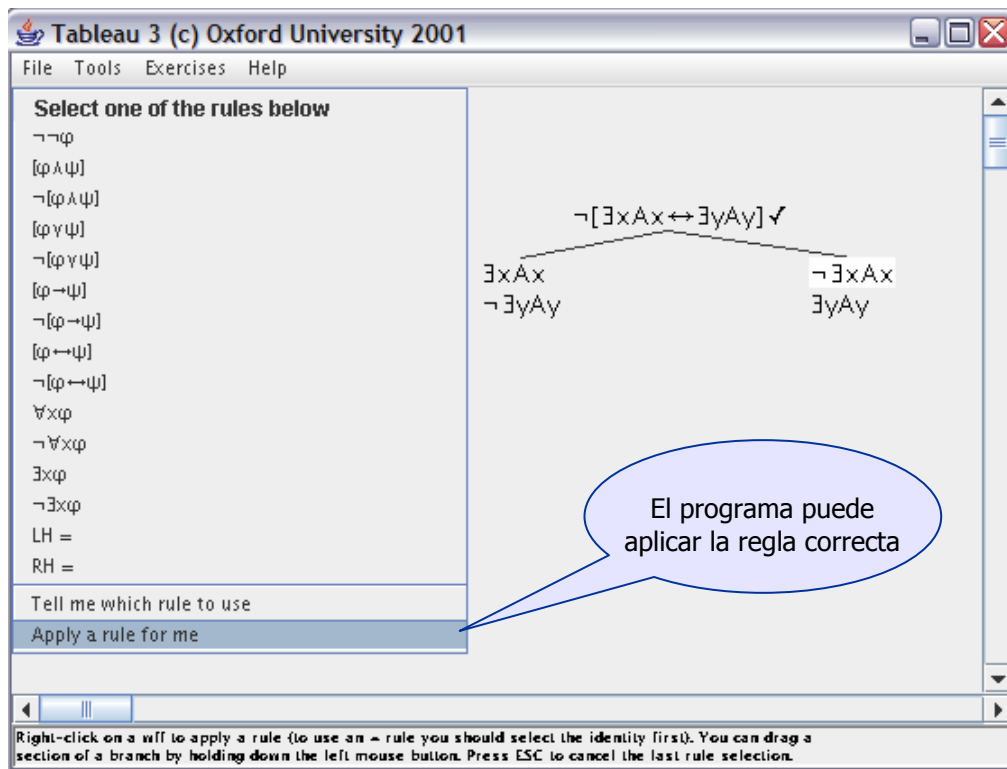


Ilustración 10. Tableau III. Funcionalidad “*Apply a rule for me*”

Una vez que no se puede aplicar ninguna otra regla al *tableau*, el programa lo indica mediante un mensaje en la barra de estado de la ventana principal, como se puede ver en la Ilustración 11

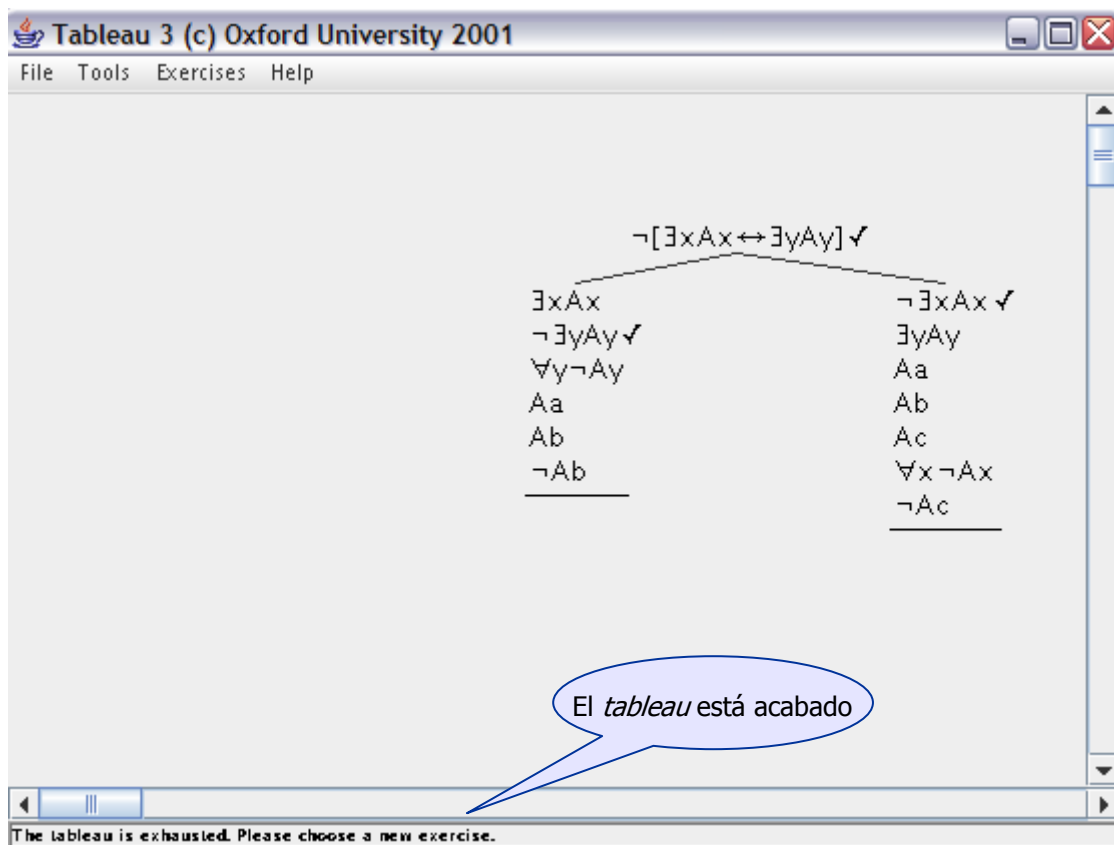


Ilustración 11. Tableau III. Ejercicio finalizado

La forma en la que se resuelven el resto de ejercicios (lógica proposicional y de predicados con identidad) es similar, por lo que no se incluyen en este documento, ya que los pasos a seguir por el usuario son los mismos. Simplemente necesita conocer las reglas y sustituciones que se pueden aplicar a cada sentencia.

3.2 Marcar (✓) las sentencias en los tableaux de predicados

Es opcional marcar las sentencias cuando se desarrollan *tableaux* de lógica de predicados. Sin embargo, **Tableau III** sigue los siguientes criterios:

- Marcar una fórmula cuando se le ha aplicado una regla proposicional
- Marcar una fórmula cuando se le ha aplicado la regla $\neg\forall x\Phi$ o $\neg\exists x\Phi$
- No marcar una fórmula cuando se le ha aplicado la regla $\neg\forall x\Phi$ o $\neg\exists x\Phi$ o la regla identidad.

Tableau III no permite reutilizar una fórmula que ha sido marcada.

3.3 Funcionalidad “User input”

Además, el Método de las Tablas Semánticas como un instrumento heurístico proporciona una forma sistemática para construir un contraejemplo.

Tableau III también considera esta opción, y permite que el usuario introduzca una fórmula. A partir de esta fórmula, la aplicación calcula el contraejemplo si es que existe.

Sin embargo, esta funcionalidad no aparece documentada ni en la ayuda de la propia aplicación, ni en la página del grupo que soporta esta herramienta. Por ello, no se le ha encontrado la utilidad que quizá merezca.

3.4 Ayuda

Tableau III ofrece distintos mecanismos de ayuda:

1. Ayuda en “tiempo real”: **Tableau III** va guiando al usuario durante el desarrollo de los *tableaux* ya que en todo momento, en la barra de estado de la aplicación aparecen mensajes indicando al usuario como debe proceder e incluso informándole de operaciones no permitidas, tales como aplicar una regla incorrecta o utilizar nombres de variables incorrectos.
2. Documentación de ayuda: **Tableau III** ofrece un mecanismo de ayuda breve que informa al usuario sobre las reglas que se pueden aplicar en los *tableaux* y como se forman fórmulas bien formadas.

4 Conclusiones

Una vez expuesta la funcionalidad que ofrece **Tableau III**, estamos en disposición de señalar sus principales ventajas y alguna desventaja.

- Ventajas
 - o El programa puede ejecutarse de dos formas distintas
 - Como una clásica aplicación de escritorio, al estar desarrollada en Java puede ejecutarse sobre cualquier sistema operativo que disponga de una máquina virtual adecuada
 - Como un Applet desde un navegador Web, de este modo no se obliga al usuario a instalar ningún software en el equipo.
 - o Mensajes de ayuda guían al usuario. Como ya se ha comentado a lo largo de este documento, el usuario encuentra apoyo por parte de la herramienta durante la construcción de los *tableaux*, ya que en todo momento se muestran mensajes explicativos e informes de errores.
 - o Distintos tipos de ejercicios que permite al usuario adquirir un conocimiento completo de todos los tipos de *tableaux* que se pueden realizar así como de las reglas que puede aplicar.
 - Proposicionales, de predicados, de predicados con identidad
- Desventajas
 - o No permite la realización de ejercicios propuestos por el usuario. Uno tiene que conformarse con trabajar con los ejercicios que hay almacenados en la herramienta
 - o Funcionalidad oscura "*User input*"
 - No figura ninguna ayuda para su uso
 - El usuario tiene que "adivinar"
 - o Ausencia de una documentación para usuarios que no tengan conocimiento exhaustivo de lógica

5 Anexo I: Requisitos del sistema

Como la herramienta está realizada en Java, puede ser ejecutada en cualquier sistema operativo que disponga de una máquina virtual de Java. Por lo tanto, la ejecución es correcta tanto en Windows como en las distribuciones de Linux o MAC OS.

Únicamente es necesario que el equipo en el que se quiere ejecutar la aplicación tenga instalado el JRE de Sun que permite la ejecución de aplicaciones Java.

Con el JRE funcionan perfectamente tanto la aplicación de escritorio como el Applet. No impone ningún otro tipo de restricción software.

6 Bibliografía

Artículo "Tablas semánticas para lógica de predicados de primer orden"

- Autor: Ángel Nepomuceno (Universidad de Sevilla)
- Fuente: <http://logicae.usal.es>

Artículo "Enseñar lógica: lenguaje lógico y árboles semánticos"

- Autor: Ángel Nepomuceno (Universidad de Sevilla)
- Fuente: <http://logicae.usal.es>

"Introduction to logic"

- Autor: Universidad de Oxford
- Fuente: <http://logic.philosophy.ox.ac.uk/>

"Logic software and education"

- Fuente: <http://www.cs.otago.ac.nz/staffpriv/hans/logiccourseware.html>