

Herramienta didáctica con Python para el aprendizaje de la lógica proposicional: LogicalPy

Didactic tool with Python for learning propositional logic: LogicalPy

Iván D. Buitrago¹, Juan E. Castaño², Santiago Giraldo³.

¹Universidad de San Buenaventura- Medellín. ivan.buitrago@usbmed.edu.co

²Universidad de San Buenaventura- Medellín. jecc2499@gmail.com

³Universidad de San Buenaventura- Medellín. santiago-giraldo-rendon@hotmail.com

 OPEN ACCESS



Copyright: © 2019 Ingenierías USBmed.
La revista *Ingenierías USBmed* proporciona acceso abierto a todos sus contenidos bajo los términos de la licencia creative commons Atribución- no comercial- SinDerivar 4.0 Internacional (CC BY-NC-ND 4.0)

Tipo de Artículo: Reflexión.
Recibido: 28-11-2018.
Revisado: 18-01-2019.
Aprobado: 28-01-2019.
Doi: 10.21500/20275846.3877

Referenciar así: I., Buitrago., J., Castaño., and S., Girado. "Herramienta didáctica con Python para el aprendizaje de la lógica proposicional: LogicalPy". *Ingenierías USBMed*, 10(1), pp.53-57, 2019.

Declaración de disponibilidad de datos: Todos los datos relevantes están dentro del artículo, así como los archivos de soporte de información.

Conflicto de intereses: los autores han declarado que no existen conflicto de intereses.

Editor: Yohana López Rivera,
Universidad de San Buenaventura,
Medellín, Colombia

Resumen. En este artículo se muestra una herramienta que permite establecer una relación entre la lógica proposicional y la lógica de programación mediante un software elaborado en Python. El usuario puede solucionar de manera didáctica ejercicios relacionados con las reglas de inferencia y por lo tanto se logra facilitar la comprensión de algunos conceptos de la lógica proposicional. El programa LogicCoach es un software basado en el libro "A concise introduction to logic" y es el punto de partida para la elaboración del software (LogicalPy) como herramienta didáctica.

Palabras Clave. Lógica; Lógica proposicional; Python, Reglas de inferencia; Programación; Deducciones.

Abstract. In this paper we introduce a tool that allows the establishment of a relationship between the propositional logic and the programming logic through a software developed in Python. The user can solve exercises related to rules of inference, in a didactic way; and therefore it is possible to ease the comprehension of some concepts of the propositional logic. The program LogicCoach is a software based on the book "A concise introduction to logic" and this is the starting point of the elaboration of the software (LogicalPy) as a didactic tool.

Keywords. Logic; Propositional logic; Python; Inferences rules; Programming; Deductions.

Nomenclatura

Reglas de inferencia tomadas en el proyecto:

1. Modus ponens (mp)
2. Método de casos (mc)
3. Adjunción (u)
4. Conmutativa (c)
5. Silogismo hipotético (sh)
6. Simplificación (s)
7. Doble negación (dn)
8. Tolendo ponens (tp)
9. Condicional (condi)
10. Definición de \wedge (def)
11. Tolendo tollens (tt)
12. Bicondicional (bic)
13. Idempotencia (id)
14. Conjunción (cc)

I. Introducción

Motivados por el deseo de encontrar diferentes formas de enseñar lógica proposicional y basados en el trabajo realizado en la Universidad de Cleveland [1], donde se enseña lógica proposicional teniendo como herramienta el programa LogicCoach, que es un software basado en el libro “A concise introduction to logic”, el cual muestra la manera de verificar la solución de ejercicios de lógica proposicional permitiendo al usuario saber hasta qué punto su solución es correcta y proporcionando sugerencias para corregir los errores presentados en la deducción. En este contexto, nuestro trabajo (LogicalPy) es una herramienta didáctica desarrollada en Python para el aprendizaje de la lógica proposicional de una forma didáctica. El programa Permite que el estudiante visualice la solución del ejercicio y todas las posibles reglas de inferencia que se pueden aplicar para obtener la solución. Es de esta forma como el educando tiene la posibilidad de ir comprendiendo el significado y la forma de aplicar cada una de las reglas de inferencia.

En cuanto a la enseñanza de la lógica, se sostiene en “Introducción a los Conceptos de la Lógica Proposicional desde la Experiencia” que en un mundo en el que la búsqueda de la verdad ha preocupado al hombre desde la antigüedad, siendo un elemento crítico en el desencadenamiento de todo tipo de conflictos, y siendo los sectores sociales hegemónicos a través de la represión o del monopolio de la comunicación quienes consiguen imponerla. Es necesario retomar la lógica como dice Maritain (citado por Togni, 2005) “Lógica es el arte que dirige el propio acto de la razón”. Luego, la lógica es ese elemento de juicio del sujeto, que le permite fundamentar las decisiones en una estructura coherente del pensamiento, que viene siendo desarrollada desde los filósofos de la antigua Grecia [2]. Es decir, la enseñanza de la lógica proposicional debe pensarse en forma estructurada, crítica y coherente para permitirle al estudiante un acercamiento

eficaz a los principios básicos de la misma.

Realizando una reflexión en cuanto a la enseñanza de la lógica proposicional, sostienen en el artículo, “la lógica proposicional y su aplicación en contexto” que los estudiantes, cuando abordan en sus cursos de matemáticas el tema de la lógica proposicional, manifiestan un alto grado de desmotivación, debido a la manera como este se les presenta. Su enseñanza se centra en encontrar el valor de verdad de las proposiciones y demostrar sus propiedades mediante el uso de tablas de verdad, olvidando, en la mayoría de los casos, la importancia de aplicar dichos conceptos y propiedades a situaciones reales, lo que daría sentido a lo que aprenden, pues además de desarrollarles el pensamiento formal, les permitiría adquirir un aprendizaje significativo [3]. Es decir, se da la interacción entre los conceptos previos vistos en clase y los nuevos conceptos adquiridos en el desarrollo del proyecto. Es claro que los conceptos relacionados con la lógica proposicional pueden ser memorizados, pero serán aprendidos e interiorizados si el maestro, en compañía de los estudiantes realiza proyectos que logren vincular los diferentes saberes, como es el caso de la lógica proposicional y la programación.

Haciendo referencia a la importancia del diseño de objetos interactivos para la enseñanza de la lógica, afirma Sergio Barrionuevo que muchas son las ventajas provistas por el uso de recursos interactivos. Primero, cuando los estudiantes pueden ver qué están haciendo y cómo funciona la semántica formal en un modelo fáctico pueden lograr una mejor comprensión de los temas estudiados. No obstante, los estudiantes tienen distintos ritmos de aprendizaje. Por lo cual, la posibilidad de utilizar recursos interactivos que pueden ser utilizados en cualquier momento, desde la web o descargados en su PC, les permite a los alumnos poder utilizarlos y reutilizarlos en cualquier momento del proceso enseñanza-aprendizaje sin necesidad de la presencia del docente, en tanto pueden repetir la explicación o actividad cuantas veces lo requieran o les resulte necesario. A lo anterior también es importante agregar que, si bien es cierto que las utilidades de los recursos interactivos son de gran ayuda en el proceso de enseñanza aprendizaje, es más significativo si el estudiante participa en la elaboración de los recursos como se evidenció en nuestro proyecto, “Herramienta didáctica con Python para el aprendizaje de la lógica proposicional: LogicalPy”.

II. Marco Teórico

La lógica proposicional se basa en las construcciones teóricas y filosóficas para determinar la veracidad de las proposiciones. Las reglas de inferencia que sustentan la lógica proposicional se encuentran claramente explicadas en texto Matemáticas Discretas [4]. En este desarrollo se realizan deducciones utilizando el método directo, que partiendo de una o varias hipótesis y con el uso de las reglas de inferencia se obtienen conclusiones válidas [4].

A. La lógica

La lógica es una rama del saber que cuenta con un recorrido histórico significativo, comparable con la geometría o la astronomía, su origen puede ubicarse en la época de reflexiones sobre la argumentación deductiva que sostenía Aristóteles y los estoicos, por los años 350 y 200 a.c.

La palabra "lógica" tiene variadas acepciones en la filosofía, la matemática, la psicología y en la programación. De acuerdo con Francisco Bonnin Aguilo, la lógica abarca cosas tan distintas como: la Lógica Aristotélica, la Lógica de los Estoicos, el arte de las disputaciones escolásticas, lógica Inductiva, Lógica Trascendente de Kant, la Lógica de Hegel, la Lógica dialéctica marxista, y la Lógica matemática con todas sus variantes [5].

Visto de esta manera es necesario precisar que la lógica explora argumentos racionales para obtener conclusiones válidas, desde la filosofía la matemática y la programación según sea el caso. En esta misma línea, pero ahondando en lo concerniente a la lógica matemática, Vicente Muñoz Delgado sostiene que la evolución de la ciencia moderna, y muy especialmente el desarrollo del pensamiento matemático, da origen a la lógica como disciplina exacta. La lógica matemática se considera hoy una importante realización de nuestro mundo cultural y su crédito aumentó por las aplicaciones computacionales y mecanismos automáticos [6].

B. Lógica de Matemática

La lógica tiene sus inicios en el paraíso de los griegos, pero cabría preguntarse en qué momento la lógica empieza a considerarse parte de las matemáticas. Gottfried Leibniz en el 1646 realiza un acercamiento valioso entre la lógica y la matemática, propone representar el conocimiento de tal manera que se pudieran obtener conclusiones válidas por medio de razonamientos mecánicos, lo que se conoce en nuestros días como lógica simbólica. Es decir, Leibniz quien además de filósofo era matemático, buscaba una forma de estructurar la lógica por medio de la matemática. Con ideas similares a las de Leibniz, George Boole muestra la estrecha relación existente entre las matemáticas y las operaciones lógicas, comparando adición, multiplicación y sustracción con la unión, intersección y la negación, dando inicio a lo que se conoce como lógica de Boole. En palabras de José Ferreirós, solemos poner el nacimiento de la lógica matemática hacia 1850, cuando George Boole algebrizó las lógicas aristotélica y estoica, a la vez que Augustus De Morgan comenzaba a investigar el razonamiento racional en matemáticas [7].

En este mismo orden de ideas, Steve Allan Russell dice que Boole ocupa un lugar preponderante en la historia de la lógica matemática, al posibilitar que su cálculo sea susceptible de una doble interpretación tanto a nivel de una lógica de clases como de una lógica sentencial. Boole fue el primero desde Aristóteles que se puso a pensar en poner letras para las afirmaciones, encontrando que el sim-

bolismo del álgebra funcionaba también para las leyes del pensamiento; poniéndole la misma notación que para los números. A lo verdadero le pone el 1 y a lo falso el 0, lo cual funciona muy bien para deducir la ley de no contradicción. El desarrollo de este cálculo se apoya en los aportes realizados por Augustus De Morgan para ampliar la silogística aristotélica, a partir del uso de letras y símbolos, como son los paréntesis y los puntos para poder escribir las distintas sentencias [8].

Es decir, los aportes de Augustus De Morgan y de George Boole pueden considerarse un punto de partida para el desarrollo de la programación. A Boole se le atribuye la invención de las tablas de verdad para comprobar la veracidad de proposiciones compuestas. Este trabajo fue retomado por Bertrand Russell y Alfred Whitehead en 1910 en su obra "Principio Matemático", quienes codificaron la lógica simbólica en su presente forma definiéndola como la "Ciencia de todas las operaciones conceptuales". Pero sin duda alguna las investigaciones que hicieron de Boole un nombre familiar entre los matemáticos e informáticos fue que siempre buscó la simplicidad interna y subyacente a los conceptos matemáticos. Le gustaba formular principios generales, expresados de forma simbólica, y dejar que los símbolos hicieran los razonamientos [9].

III. Metodología

Se tiene como base para la construcción de LogicalPy, el software LogicCoach que utiliza los conceptos desarrollados por Patrick J. Hurley [1]. Este software se utiliza para determinar la validez de las deducciones de problemas de lógica proposicional, mostrando los errores en el procedimiento y en general se usa como herramienta para practicar y verificar la manera correcta de utilizar las reglas de inferencia. Quizás, la mayor dificultad presentada en el uso de esta herramienta se debe a que la sintaxis es difícil de entender y se requiere un conocimiento previo del texto de Patrick J. Hurley [1]. Además, la manera como se definen las reglas de inferencia es compleja, pues debe especificarse con demasiado detalle el operador a utilizar y el orden predeterminado. El LogicCoach no es muy flexible a la hora de simplificar ciertos pasos que se pueden omitir en el apartado de la revisión. La siguiente gráfica representa el funcionamiento del programa LogicCoach.

Con el objetivo de comparar y desarrollar una nueva herramienta, primero desarrollamos los pasos utilizados en el LogicCoach para obtener los resultados. Esta herramienta verifica el cumplimiento de las reglas de inferencia en cada uno de los procedimientos realizados siguiendo el esquema presentado en la Figura 1.

Se ingresa a la pantalla de inicio, luego se busca el ejercicio a ejecutar y se resuelve de forma manual, después se ingresa la solución para que el software verifique, si el ejercicio está bien resuelto el software muestra "OK", en caso contrario señala la regla de inferencia que fue mal utilizada y deja ver una ejemplificación. Se repite el proceso

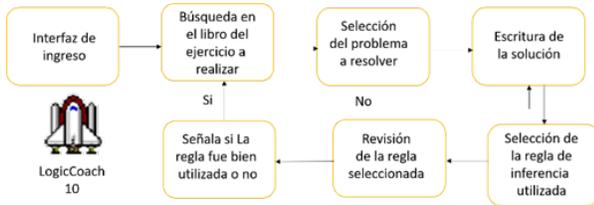


Figura 1. Diagrama de flujo del LogicCoach

hasta terminar la solución del ejercicio. Es evidente que el usuario debe conocer el texto, “A concise introduction to logic”, y tener conocimientos avanzados de las reglas de inferencia.

A partir de las dificultades mencionadas anteriormente y basados en los conceptos vistos la clase de matemáticas discretas [4] y fundamentos de programación [10], se creó un software en Python teniendo como punto de partida el presentado por LogicCoach.

En principio se tomaron ciertos aspectos del LogicCoach y se implementaron de tal manera que el software creado solucionara ejercicios propuestos por el usuario utilizando las reglas de inferencia, pero mostrando en cada paso de la demostración las reglas utilizadas. Además, permite obtener todas las posibles reglas de inferencia aplicables a las hipótesis que se tienen y así llegar a la deducción indicada. Esto se logró desarrollando la programación en Python de cada una de las reglas de inferencia utilizadas en la lógica proposicional [2] por medio de las estructuras básicas de programación como ciclos, matrices, listas y condiciones. En un principio se implementaron algunas reglas de inferencia utilizando funciones o definiciones (como son llamadas en Python) las cuales son porciones de código que se encuentran por fuera del método principal ejecutable y pueden ser llamadas cuando sea necesaria su utilización, al respecto afirma Paul Deitel y Harvey Deitel en el libro “java cómo programar”:

“Los métodos (también conocidos como funciones o procedimientos en otros lenguajes) permiten al programador dividir un programa en módulos, por medio de la separación de sus tareas en unidades autónomas.

Una razón para dividir un programa en módulos mediante los métodos es el enfoque “divide y vencerás”, que hace que el desarrollo de programas sea más fácil de administrar, ya que se pueden construir programas a partir de piezas pequeñas y simples” [11].

En la Figura 2 se describe gráficamente el procedimiento utilizado para el funcionamiento del LogialPy. Inicialmente se ingresan las hipótesis una a una y se interpreta cada una para poder llegar a relacionarlas entre sí, a continuación, se ofrecen las posibles reglas de inferencia a aplicar y posteriormente se muestra la deducción requerida, permitiendo al estudiante novato familiarizarse con la lógica proposicional.

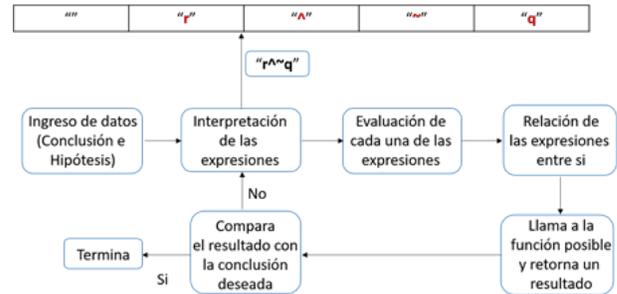


Figura 2. Diagrama de flujo del LogicalPy.

IV. Análisis de Resultados

Al realizar las diferentes pruebas se pudieron reproducir las deducciones de alrededor de 50 ejercicios de lógica proposicional propuestos en los textos [1] [4]. Esto muestra que el software es confiable y que permite reproducir ejemplos y ejercicios que se enseñan en los cursos de matemáticas discretas, con lo cual se puede evidenciar el aporte a los procesos de enseñanza- aprendizaje de este curso.

```
Deducir:
p
hipotesis:
1 . q>p
2 . qv5
3 . ~s
solucion:
4 . ~qvp          condi en 1
5 . ~(~q^~s)     def en 2
6 . q            tp en 2 y 3
7 . ~q>s         condi en 2
8 . svq         C en 2
9 . ~s^q        cc en 3 y 6
10 . ~svu       u en 3
11 . p          mp en 1 y 6
```

Figura 3. Deducción proposicional.

Un ejemplo de deducción se presenta en la Figura 3, en la cual se muestra el proceso realizado por el software para generar la deducción, así como cada una de las reglas de inferencia utilizadas.

V. Conclusiones

Se evidencia un software sólido para la solución de ejercicios con la utilización de reglas de inferencia básicas, por el método directo. Además, es importante anotar que el desarrollo del software permitió la conexión entre la lógica proposicional y la lógica de programación.

Sumado a lo anterior, también se puede afirmar que LogicalPy permite al estudiante del curso de matemáticas discretas, entender con mayor claridad las reglas de inferencia, siendo una herramienta didáctica que acompañe y fortalezca los procesos académicos. Los estudiantes del curso de matemáticas discretas se mostraron más receptivos cuando vieron la posibilidad de conjugar la lógica proposicional del curso de matemáticas discretas con los contenidos del curso de lógica de programación.

VI. Trabajos futuros

Para un próximo proyecto se realizará una ampliación del código que permita la implementación de una IA (inteligencia artificial) de tal manera que se pueda resolver cualquier tipo de ejercicio.

Se pretende crear software semejante al realizado en este proyecto, pero elaborado en Java, de tal forma que pueda tener una interfaz gráfica para los usuarios. Además de lo anterior se sugiere pensar en la creación de una aplicación móvil que le permita al estudiante tener los contenidos a la mano en cualquier momento y en cualquier lugar.

Referencias

- [1] P. Hurley, *A concise introduction to logic*. Cengage Learning, 2005.
- [2] J. C. Gutierrez Vanegas *et al.*, *Introducción a los Conceptos de la Lógica Proposicional desde la Experiencia*. PhD thesis, Universidad Nacional de Colombia-Sede Manizales.
- [3] S. A. Alarcón, C. M. Restrepo, and H. J. Herrera, “La lógica proposicional y su aplicación en contexto,” 2015.
- [4] C. Gaviria, C. Márquez, and I. Buitrago, *Matemáticas Discretas*. Editorial Bonaventuriana, 2015.
- [5] F. Bonnin, “Ei concepto de la lógica a lo largo de la historia de la filosofía,” *Recuperado el*, vol. 2, 1982.
- [6] V. Muñoz Delgado, “Consideraciones sobre la lógica y su historia,” 1979.
- [7] J. M. Ferreirós Domínguez, “La lógica matemática: una disciplina en busca de encuadre,” *Theoria. Revista de Teoría, Historia y Fundamentos de la Ciencia*, 25 (3), 279-299, 2010.
- [8] S. Allan Russell, “Tránsitos conceptuales en la lógica de bertrand russell de 1900 a 1927,” Master’s thesis, Facultad de Filosofía.
- [9] I. Stewart, “Mentes maravillosas: Los matemáticos que cambiaron el mundo,” 2018.
- [10] L. Joyanes Aguilar, “Fundamentos de programación: algoritmos, estructura de datos y objetos/por luis joyanes.”
- [11] P. J. Deitel and H. M. Deitel, *Java: cómo programar*. Pearson, 2012.