EL (GRAN) JUEGO DE LA LÓGICA #1

Juego

La lógica es muchas cosas[1], pero nosotros la vamos a considerar un juego. Porque divierte. Pero, además, porque:

1) Tiene objetivos propios…
2) Que se logran mediante reglas propias…
3) Y “fichas” propias

Y es un GRAN juego porque no es *uno*, sino *varios* juegos en uno. Con la diferencia favorable de que todos ellos están relacionados. Vamos a jugar a este juego porque nos servirá como entrenamiento mental para las principales cuestiones filosóficas, que veremos el resto del curso

Fichas: es lo primero que necesitamos, aparte de jugadores (que van a ser vds.). O piezas, o materiales. Para abaratar costes, nuestras fichas van a ser **símbolos**. Escritos. Fáciles. Algunos, comunes a otros juegos; otros, específicos del nuestro. Los que yo diga, que para eso me estoy inventando yo el juego. Más tarde veremos qué hacemos con ellos.

Tabla de símbolos

No son muchos, pero los vamos a ordenar en diversas categorías, porque se utilizarán de diferentes formas. Recuerden que tanto su forma como su nombre son arbitrarios, convencionales y no significan nada (de momento):

1. Unos cuantos que llamaremos “variables”: **p**, **q**, **r**, **s** … y otras letras minúsculas sucesivas, si fueran necesarias (no se incluyen las comas en esta ni en las otras categorías)
2. Cinco más que llamaremos “operadores” . Para complicar un poco las cosas, habrá dos tipos de operadores:
   a. Operadores “diádicos”: ∧, ∨, →, ↔ Esos cuatro. Ni uno más, ni uno menos: una especie de uve, una especie de capirucho, una flecha y una flecha doble. Porque sí.
   b. Operador “monádico: ¬ Uno solo, ¿pasa algo?
3. Símbolos auxiliares (se llaman así porque auxilian, ayudan. Son relativamente menos importantes, pero también los necesitamos). Otros dos tipos:
   a. “Metavariables”: **P**, **Q**, **R**, **S** … Exacto, como las variables, pero en mayúsculas. Y su nombre también parece decirnos algo ¿verdad? Pues no. No es eso. Es porque sí. Es un nombre como otro cualquiera.
   b. Otros cuantos símbolos variados que no tienen ni siquiera nombre. Estos: (,), 1, 2, 3 …, ———, y alguno más.

Primer juego: Fórmula bien formada (fbf)

Llamaremos “fórmula” a cualquier serie o hilera o sarta de símbolos, de cualquier longitud, de entre los que hemos enumerado más arriba. Por ejemplo:

p→ ∨ r (

es una fórmula. Ahora bien, a partir de este momento, *no vale cualquier fórmula*: sólo aceptaremos determinadas fórmulas, y rechazaremos todas las demás. Así, que será ese nuestro primer juego: distinguir “fórmulas bien formadas” (fbf’s) de “fórmulas mal formadas” (fmf’s, como la del ejemplo, por cierto) o, alternativamente, saber construir fbf's. Una fórmula estará bien formada si, en su

[1] Pero es también, y de un modo que quedará claro más tarde, un lenguaje, con su sintaxis y su semántica: sus signos, sus reglas gramaticales, su capacidad significante etc...

construcción, se ha seguido alguna o varias (una o varias veces) de las siguientes **reglas de formación** (o “reglas sintáticas”):

1. Una **variable sola** es una fbf
2. El **operador monádico seguido de UNA fbf** es una fbf
3. **UN operador diádico interpuesto entre DOS fbf’s** es una fbf

Son sólamente tres las reglas de formación. Sólo hay tres maneras de fabricar fbf's, pero, como se pueden repetir indefinidamente, pueden escribirse fbf's de cualquier complejidad y longitud. Fíjense en que:

* **pq** NO es una fbf, porque ninguna de las reglas autoriza a pegar, por todo el morro, dos variables. O, visto de otra forma, ninguna regla autoriza a unir dos fbf's sin más.

* **Q** NO es una fbf, porque ninguna regla autoriza el uso de metavariables (se usarán para otra cosa, no para construir fbf's)

* **p ¬** NO es una fbf, porque el operador monádico **debe preceder** a otra fbf para constituir una nueva fbf, no seguirla

* **p r →** NO es una fbf, porque el operador diádico **debe intercalarse** entre las dos fbf's

Argumentos y operadores

En general, cuando haya que decidir sobre la buena o mala formación de una fórmula compleja, habrá que ir analizándola por partes, sabiendo que a) **todo operador MONÁDICO debe tener a su derecha UNA fbf** (mejor dicho, una serie de símbolos que *serían* una fbf si estuviesen solos) y b) **todo operador DIÁDICO debe encontrarse entre DOS fbf's** (o, mejor dicho, dos series de símbolos que *serían* fbf's si estuviesen, por separado, solas). Después del recuento, no debe sobrar ni faltar nada.

Llamaremos “**argumento**” a esa fbf que debe seguir al operador monádico para constituir una nueva fbf y a cada una de las dos fbf's entre las cuales se sitúa un operador diádico. Así, todo operador monádico debe tener clara y correctamente definido **su argumento** y todo operador diádico debe tener clara y correctamente definidos **sus dos argumentos** para que la fórmula completa sea una fbf.

Y llamaremos **operador principal** de una fbf a aquel operador cuyo(s) argumento(s) constituye(n) la totalidad de la fórmula.

Veamos estos casos:

- **¬ p** FBF, porque el operador monádico se encuentra a la izquierda de su argumento (la variable **p**, que era, por la primera regla, una fbf)
- **¬ ¬ p** FBF también. La segunda regla *no* dice que se unan dos operadores monádicos, pero, con todo, es una fbf. Cada operador tiene definido perfectamente su argumento: el argumento del segundo operador es la variable **p** (que sería una fbf, por la primera regla); ¿el argumento del primer operador está definido correctamente? Sí, es el grupo ¬ **p**, que, según acabamos de ver, sería también una fbf si estuviese aislado. Consiguientemente, podemos acumular ilimitadamente operadores monádicos delante de una variable y el resultado seguirá siendo fbf[2].
- **p → ¬r** FBF: De nuevo, no se dice nada de unir dos operadores en las reglas, pero esta unión resulta de la aplicación de las mismas. El operador flecha, como ordena la tercera regla, se interpone entre dos grupos de símbolos: la variable **p** (que era una fbf) y el grupo ¬**r** (que era, de acuerdo con la segunda regla, otra fbf)
- **p → ∨ r** FMF. Repasa cada operador, y verás que falta o sobra algo: ¿Cuáles son los argumentos de la flecha? El primero está bien definido (la variable **p**), pero, ¿y el segundo? El grupo ∨ **r** NO sería una fbf en ningún caso, luego tampoco puede serlo el conjunto. Nunca habrá dos (o más) operadores diádicos juntos

[2] Esto no quiere decir que todas esas fbf’s resultantes sean idénticas: seguirán siendo fbf, pero distintas entre sí.

Símbolos auxiliares

- **$p \rightarrow q \vee r$** Aquí hay que tener otra cosa en cuenta. ¿Cada operador tiene bien definidos sus argumentos? El primero de la flecha es, sin duda, la variable **p**. Pero, ¿cuál es el segundo? Como un argumento es la serie de símbolos que constituirían fbf si estuviesen solos, resulta que hay dos posibilidades: podría ser la variable **q**, pero también podría serlo el grupo **$q \vee r$**. Este juego (la lógica entera) es absolutamente exigente, por lo que hemos de decidir **sin ambigüedad** cuáles son los argumentos. En este caso, sería necesario recurrir a los paréntesis, exactamente igual que se utilizan en aritmética. Debemos escribir, para que se considere fbf genuina, o **$p \rightarrow (q \vee r)$**, en cuyo caso el operador principal es la flecha, o **$(p \rightarrow q) \vee r$**, en cuyo caso el operador principal es la $\vee$. Aunque estas dos fórmulas no son la misma, ni son equivalentes, ambas estarían bien formadas.

De modo que, cuando sea necesario, recurriremos al uso de paréntesis (o corchetes, llaves etc.) para especificar sin ambigüedades el "alcance" de cada operador. Pero sólo cuando sea preciso. Esto, por ejemplo, es exagerado e innecesario:

$$p \rightarrow (\neg (\neg p))$$

puesto que, sin los paréntesis, siguen siendo inequívocos los argumentos de cada operador.

En general, nunca usaremos paréntesis cuando su eliminación no origine ambigüedades. En particular, **nunca los usaremos para englobar dos o menos símbolos**. Cuando dos o menos símbolos deban entenderse agrupados, simplemente omitiremos los paréntesis, y los reservaremos para el caso de que queramos expresar otro reparto. Por ejemplo, en

$$\neg p \rightarrow q$$

debemos entender que el operador monádico alcanza sólo a la variable **p**, es decir, debemos entender **$(\neg p) \rightarrow q$**, y que el operador principal es la flecha. Si quisiéramos expresar que el operador principal es el monádico y que la flecha conecta sólo las variables **p** y **q**, habríamos de usar los paréntesis, tal que así:

$$\neg (p \rightarrow q)$$

Así, pues, **uno debe estar en condiciones, para este juego y para todos los demás de la lógica, de reconocer el alcance de cada operador**. Esto es fundamental. FUN-DA-MEN-TAL. Tanto, que lo repito: **<u>uno debe estar en condiciones, para este juego y para todos los demás de la lógica, de reconocer el alcance de cada operador</u>:**

$$[(p \rightarrow q) \wedge p] \rightarrow q$$

Veamos. Cada operador debe tener su (si es monádico) o sus (si es diádico) argumentos, sin que sobre ni falte nada. Además, los paréntesis (o corchetes, como en este caso) deben deshacer toda ambigüedad.

- Tomemos el primer operador, la flecha: sus argumentos son 1º la **p** y segundo sólo la **q**, porque el paréntesis delimita el alcance.
- El siguiente operador es el capirucho ese, cuyo primer argumento es el grupo **$(p \rightarrow q)$** y el segundo es sólo la siguiente **p** (porque el paréntesis fija el alcance del capirucho).
- Finalmente, el último operador flecha –que es también el operador principal– conecta, como primer argumento, el grupo **$(p \rightarrow q) \wedge p$** (los corchetes delimitan el alcance) y, como segundo, la **q**. Esto es vergonzantemente fácil, pero veamos otro.

$$[(p \rightarrow q) \wedge (q \rightarrow r)] \leftrightarrow (p \vee \neg p)$$

¿Qué me dicen? Les anticipo que es una fbf. Vean si logran identificar todos los argumentos de cada uno de los operadores.