Operador Lógico	Negação	Conjunção	Disjunção	Condicional	Bicondicional	
Leitura - Símbolo	NÃO ~	ΕΛ	OU V	SE ENTÃO →	SE E SOMENTE SE	\leftrightarrow

Logica Formal ou Simbolica - Logica das Proposições (Para analizar as formas de argumento é preciso realizar a formalizacao de enunciados ou proposições) isto é usar letras setenciais e operadores logicos.

Equivalencias Logicas

Uma equivalência lógica é:

Reflexiva: P≡ P

Simétrica: se P≡ Q, então Q≡ P

Transitiva:

se P≡ Q e Q≡ R, então P≡ R

Dupla Negação (DN)

P ≡ ~~P

Tautologia (TAUT)

 $P \wedge P \equiv P$ $PVP \equiv P$

Comutação (COM)

 $PVQ \equiv QVP$ $P \wedge Q \equiv Q \wedge P$

Associação (ASSOC)

 $PV(QVR) \equiv (PVQ)VR$ $P \wedge (Q \wedge R) \equiv (P \wedge Q) \wedge R$

Distribuição (DIST)

 $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$ $PV(Q\Lambda R) \equiv (PVQ)\Lambda(PVR)$

Absorção (ABS)

 $P \wedge (P \vee Q) \equiv P$ $PV(P\Lambda Q) \equiv P$

Leis de DeMorgan (DM)

 \sim (P \wedge Q) \equiv \sim P V \sim Q \sim (P V Q) \equiv \sim P \wedge \sim Q

Implicação Material (IM)

 $P \rightarrow Q \equiv P \vee Q$

Transposição (TRANS)

 $P \rightarrow Q \equiv \sim Q \rightarrow \sim P$

Exportação (EXP) $(P \land Q \rightarrow R \equiv P \rightarrow Q \rightarrow R)$

Bicondicional (BICOND)

 $P \leftrightarrow Q \equiv (P \rightarrow Q) \land (Q \rightarrow P)$

 $P \leftrightarrow Q \equiv (\sim P \land \sim Q) \lor (P \land Q)$

Regras de inferência

Premissas - Conclusão

Dupla Negação (DN)

~~P

Introdução de Λ (ΛΙ) P. Q $P \wedge Q$

Eliminação de " A " (A E)

 $P \wedge Q$ P (ou Q)

Introdução de "V" (VI) PVQ

Modus Ponens (MP)

 $P \rightarrow Q. P$

Modus Tolens (MT)

P→ Q, ~Q

Dilema Construtivo (DC)

 $P V Q. P \rightarrow Z. Q \rightarrow Z$

Introdução de bicondicional (↔I)

 $P \rightarrow Q, Q \rightarrow P$ P↔Q

Eliminação de bicondicional (↔E) P↔Q $P \rightarrow Q \land Q \rightarrow P$

Silogismo Disjuntivo (SD) P V Q, ~Q

Silogismo Hipotético (SH)

 $P \rightarrow Q. Q \rightarrow R$ $P \rightarrow R$ Absorção (ABS)

 $P \rightarrow Q$ $P \rightarrow (P \land Q)$

Sentenças Declarativas

São frases na linguagem natural utilizadas para informar ou declarar um fato. Podem ser afirmativas ou negativas.

Proposições

Uma proposição ou enunciado é um significado ou ideia expressável por uma sentença (sequência de palavras) declarativa.

Valor verdade

Cada proposição possui um valor verdade, que ou é verdadeiro ou é Falso. Nenhuma proposição possui mais de um valor verdade. Na expressão P→ Q , P é chamado

"antecedente", e Q é chamado "consequente".

bicondicional

* So É F quando o antecedente for V e o consequente F". P é condição suficiente para Q Q é condição necessária para P

ordem de precedencia

Major precedência: ~ Precedência intermediária: Λ. V Menor precedência: →. ↔

(2elevado a n) n = número de letras sentenciais, "PΛQ", que possui 2 letras sentenciais, a tabela verdade é composta por 2² = 4 linhas

TAUTOLOGIA = valores-verdade sempre verdadeiros. CONTRADIÇÃO = valores-verdade sempre falsos. CONTINGÊNCIA = valores-verdade verdadeiros e falsos

Consequência Lógica

Denomina-se premissas (ou hipóteses) certas proposições que são usadas para chegar a uma outra proposição denominada conclusão (ou tese), baseada em um raciocínio. Essas proposições compõem um argumento.

se as premissas forem verdadeiras, então a conclusão também deve ser verdadeira.

Essas proposições que formam um argumento podem ser representadas utilizando o traço de asserção | para separar as premissas de sua conclusão

Se chover, então a rua ficará alagada. (premissa) C →R Choveu. (premissa) C

Portanto (conclusão), a rua ficou alagada. R

Essa consequência lógica pode ser representada da sequinte forma:

 $C \rightarrow R, C \mid R$

Indicadores de premissas: pois, desde que, como, porque, assumindo que, visto que, supondo que, em vista de, dado que, sabendo-se que, como consequência de, etc.

Indicadores de conclusão: portanto, por consequinte. assim, dessa maneira, neste caso, daí, logo, de modo que, então, assim sendo, consequentemente, o (a) qual implica que, etc

Demonstração Condicional e Indireta

* As duas próximas regras são denominadas regras hipotéticas, pois empregam o raciocínio hipotético, que é um raciocínio baseado em hipóteses, isto é, uma suposição feita sobre o argumento a fim de mostrar que uma conclusão particular segue daquela suposição.

Prova do condicional (PC): usada quando a conclusão é uma proposição condicional.

Exemplo - Prove a forma do seguinte argumento:

```
P \rightarrow Q, Q \rightarrow R \vdash P \rightarrow R
Solução:
1 P \rightarrow Q P
2 Q→R P
3
             H (Hipótese)
    Q
             1.3 MP
5
    R
             2.4 MP
6 P→R 3.5 PC
```

Redução do absurdo (RAA): dada uma derivação de uma contradição a partir de uma hipótese X, podemos descartar a hipótese e inferir ~X.

Exemplo - Prove a forma do seguinte argumento:

```
P→Q, ~Q | ~P
Solução:
1 P \rightarrow Q P
2 ~Q P
3 | P
        H (Hipótese)
4
   Q
       1,3 MP
   Q A ~Q (contradição) 2,4 AI
       3,5 RAA
  `~P
```

VALIDADE DE ARGUMENTOS MEDIANTE REGRAS DE INFERÊNCIA

Um método eficiente de estabelecer a validade de um argumento muito extenso é deduzir a sua conclusão a partir de suas premissas, mediante uma seguência de raciocínios elementares, dos quais se saiba que cada um é válido, como as regras de inferência e equivalências lógicas. Esse processo denomina-se sequência de demonstração ou

No caso da tabela-verdade, essa pode ser construída para formas de argumento, como é o caso do exemplo abaixo. Nesse caso, para o argumento ser logicamente válido, a consequência lógica deve ser uma tautologia, isto é, NÃO pode haver uma linha da tabela em que as premissas são verdadeiras e a conclusão é falsa. No exemplo, a forma de argumento é válida, pois na linha em que ambas premissas "PvQ" e "~P" são verdadeiras (terceira linha), a sentença "Q" também é verdadeira.

```
Exemplo: P v Q, ~P | Q
P Q PvQ
V V
      V
                     V
V F
      V
                     F
F V
      V
                     ٧
FF
       F
p ai = c
t cd = ~a
```

^{*}a=antecedente *c=consequente *ponens *tolens