Let p and q be propositions (mệnh đề)

=>

+) p is *True* => “¬p” is False

+) The *conjunction* of p and q “p∧q” (p and q) is *True* when both *q and p are True*, and is *False* otherwise

|  |  |  |
| --- | --- | --- |
| p | q | p ∧ q |
| T | T | T |
| T | F | F |
| F | T | F |
| F | F | F |

+) The *disjunction* of p and q “p∨q” (p or q) is *False* when both *q and p are False,* and is *True* ortherwise

|  |  |  |
| --- | --- | --- |
| p | q | p ∨ q |
| T | T | T |
| T | F | T |
| F | T | T |
| F | F | F |

+) The *exclusive* or of p and q “p ⊕ q” is *True* when *exactly one of p and q is True,* and is *False* otherwise

|  |  |  |
| --- | --- | --- |
| p | q | p ⊕ q |
| T | T | F |
| T | F | T |
| F | T | T |
| F | F | F |

+) The *conditional statement* “p → q” (*if p, then q*) is *False* when *p is True* and *q is False*, and is *True* otherwise.

In the conditional statement p → q, p is called the hypothesis (or antecedent or premise) and q is called the conclusion (or consequence).

|  |  |  |
| --- | --- | --- |
| p | q | p → q |
| T | T | T |
| T | F | F |
| F | T | T |
| F | F | T |

+) The *biconditional statement* “p ↔ q” (*p if and only if q*) is *True* when *p and q have the same truth values* ( *q and p*  are both *True* / *q and p* are both *False*), and is *False* otherwise.

Biconditional statements are also called bi-implications

|  |  |  |
| --- | --- | --- |
| p | q | p ↔ q |
| T | T | T |
| T | F | F |
| F | T | F |
| F | F | T |

|  |  |  |
| --- | --- | --- |
| Statement | When True? | When False? |
| ∀x ∀y P(x, y)  ∀y ∀x P(x, y) | P(x, y) is true for every pair x, y | There is a pair x, y for which P(x, y) is false |
| ∀x ∃y P(x, y) | For every x these is a y for which P(x, y) is true | There is an x such that P(x, y) is false for every y |
| ∃x ∀y P(x, y) | There is an x for which P(x, y) is true for every y | For every x there is a y for which P(x, y) is false |
| ∃x ∃y  ∃y ∃x | There is a pair x, y for which P(x, y) is true | P(x, y) is false for every pair x, y |

Bởi vì các câu lệnh điều kiện đóng một vai trò thiết yếu như vậy trong lập luận toán học, một loạt các thuật ngữ được sử dụng để biểu thị P → q. Bạn sẽ gặp hầu hết nếu không phải tất cả các cách sau để diễn đạt câu lệnh điều kiện này:

"if p, then q" hoặc "if p, q" hoặc "p only if q."

"p là đủ đối với q" hoặc "điều kiện cần đối với p là q"

"điều kiện đủ cho q là p" hoặc "q bất cứ khi nào p"

"q nếu p" hoặc "q khi p"

"điều kiện cần thiết cho p là q" hoặc "q trừ khi ¬p"

"p ngụ ý q" hoặc "p chỉ khi q"

"q là cần thiết cho p" hoặc "q theo sau từ p."





