



First Order Logic

Next Topic:

Validity of FOL Expressions Involving Implication

Website : <https://www.goclasses.in/>

$$\forall_{\alpha} P(\alpha) \xrightarrow{\checkmark} \exists_{\alpha} P(\alpha) \equiv \text{Valid}$$

Domain: make $\forall \alpha P(\alpha) = \text{True}$; Can you
make $\beta = \text{false}$

a — $P = \text{True}$

b — $P = \text{True}$

c — $P = \text{True}$

;

Already $P(q) = \text{True}$

$\alpha, \beta = \text{FOL Expressions}$



To make it false

$\alpha = \text{True}$ and
 $\beta = \text{false}$

Simultaneously

Procedure: $\alpha, \beta = \text{FOL Expression}$

$$(\alpha) \rightarrow (\beta) : \underline{\text{valid ??}}$$

Step 1: Assume Domain $D = \{a, b, c, d, \dots\}$

Step 2: Try to make $\alpha \rightarrow \beta$ false

o R App 1: $\alpha=T \rightarrow \beta=F$
App 2: $\beta=F \rightarrow \alpha=T$

If we "can" make $\alpha \rightarrow \beta$ false

then $\alpha \rightarrow \beta$ is Invalid.

If we "cannot" make $\alpha \rightarrow \beta$ false,

then $\alpha \rightarrow \beta$ is Valid.

$$\mathcal{E} \ell : \forall_{\delta x} \left(P_{(\alpha)} \wedge Q_{(\alpha)} \right) \xrightarrow{\quad ? \quad} \forall_{\alpha} P_{(\alpha)} \wedge \forall_{\delta x} Q_{(\alpha)}$$

Approach 1: make $\alpha = \text{True}$

then Try to make $\beta = \text{false}$.

Domain: $\forall_n (P_{(n)} \wedge Q_{(n)}) = \text{True}$

$$q = \underbrace{P \wedge Q}_{=} = T$$

$$b = \underbrace{P \wedge Q}_{=} = T$$

$$c = \underbrace{P \wedge Q}_{=} = T$$

$$a \Rightarrow P=T, Q=T$$

$$b \Rightarrow P=T, Q=T$$

$$c \Rightarrow P=T, Q=T$$

$$\frac{\forall_n P_{(n)} \quad \wedge \quad \forall_n Q_{(n)}}{T} \\ \downarrow \quad \quad \quad \downarrow \\ \text{True} \wedge \text{True}$$

$$= \text{True}$$

Cannot make
P false.



So

$$\forall n (P_{(n)} \wedge Q_{(n)}) \rightarrow ((\forall n P_{(n)}) \wedge (\forall n Q_{(n)}))$$

valid

$$\forall_{\alpha} (P_{(\alpha)} \wedge Q_{(\alpha)}) \leftarrow (\forall_{\alpha} P_{(\alpha)}) \wedge (\forall_{\alpha} Q_{(\alpha)})$$

β

make $\alpha = \text{True}$

Can we make $\beta = \text{false}$?

$$a \Rightarrow P(a) \wedge Q(a) = \text{True}$$

$$b \Rightarrow P(b) \wedge Q(b) = \text{True}$$

c \Rightarrow

$$d \Rightarrow$$

$P = T$	$Q = T$	$\in q$ <small>Domain:</small>
$P = T$	$Q = T$	$\in b$
$P = T$	$Q = T$	$\in c$
$P = F$	$Q = T$	\vdash



So After making $\alpha = \text{True}$, Cannot make $\beta = \text{false}$,

So $\alpha \rightarrow \beta$ is valid.

$$\forall x ((P(x) \wedge Q(x)) \leftrightarrow (\forall x P(x) \wedge \forall x Q(x)))$$



So After making $\alpha = \text{True}$, Cannot make $\beta = \text{false}$,

So $\alpha \rightarrow \beta$ is valid.

$$\forall x ((P(x) \wedge Q(x)) \equiv \forall x P(x) \wedge \forall x Q(x))$$



$$\forall_n (P(n) \wedge Q(n))$$

Scope of \forall_n

$$a : P(a) \wedge Q(a)$$

$$b : P(b) \wedge Q(b)$$

$$c : P(c) \wedge Q(c)$$

$$(\forall_n P(n)) \wedge (\forall_n Q(n))$$

$$\left. \begin{array}{l} P(a) \\ P(b) \\ P(c) \end{array} \right\}$$

$$\left. \begin{array}{l} Q(a) \\ Q(b) \\ Q(c) \end{array} \right\}$$



An example of a valid formula is $\forall x(P(x) \wedge Q(x)) \Rightarrow \forall yP(y)$. An example of an unsatisfiable formula is $\forall xP(x) \wedge \exists y\neg P(y)$.



$$\forall_{\alpha} (P(\alpha) \wedge Q(\alpha))$$

Domain: α

$$a \Rightarrow P(a) \wedge Q(a) \text{ make } \alpha = \text{True}$$

$$b \Rightarrow P(b) \wedge Q(b) = \text{True}$$

$$c \Rightarrow P(c) \wedge Q(c) = \text{True}$$

$$\vdots \Rightarrow P(c) \wedge Q(c) = \text{True}$$

$$\stackrel{?}{\checkmark} \rightarrow \forall_y P(y) : \underline{\text{Valid}}$$

β

Can we make
 $\beta = \text{false}$? $\Rightarrow \text{No}$
 Already, for all,
 $P = \text{True}$.

$$\forall_{\alpha} (\rho_{(\alpha)} \wedge \varrho_{(\alpha)})$$

β

Can you make β false?

$$a \Rightarrow \rho_{(a)} = T; \varrho_{(a)} = F$$

$$\underline{\rho_{(a)} \wedge \varrho_{(a)} = \text{false}}$$

$$\forall_y \rho_{(y)}$$

; Invalid

make $\alpha = \text{True}$

$$\left. \begin{array}{l} a \Rightarrow \rho_{(a)} = T \\ b \Rightarrow \rho_{(b)} = T \\ c \Rightarrow \rho_{(c)} = T \\ \vdots \end{array} \right\} \text{Domain}$$

G :

$\forall x P(x)$

\equiv

$\exists y \neg P(y)$

for everyone
for someone

P = True

P = false

and

can never happen



First Order Logic

Next Topic:

Logical Equivalences Involving Quantifiers

Website : <https://www.goclasses.in/>



$\alpha, \beta = \text{FOL expressions}$

If
and

$$\begin{array}{ccc} \alpha & \longrightarrow & \beta \\ \beta & \longrightarrow & \alpha \end{array}$$

= Valid
= Valid

}

$$\alpha \equiv \beta$$

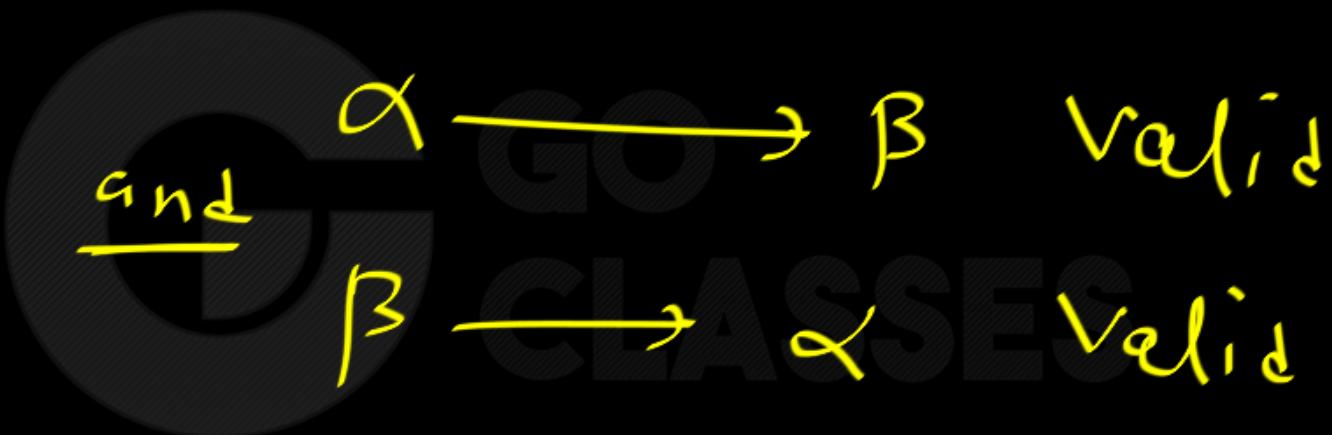
so

$$\alpha \leftrightarrow \beta = \text{Valid}$$



$$\alpha \equiv \beta$$

iff





Equivalences in Predicate Logic

- Statements involving predicates and quantifiers are *logically equivalent* if and only if they have the same truth value for...
 - every predicate substituted into these statements and
 - every domain used for the variables in the expressions.
- The notation $S \equiv T$ indicates that S and T are logically equivalent.
- **Example:** $\forall x \neg \neg S(x) \equiv \forall x S(x)$



$$\forall_{\delta \in P(x)} \equiv \exists_{\exists_{\delta \in P(x)}} \checkmark$$



GO
CLASSES



$$\forall_n (P_{(n)} \wedge Q_{(n)}) \equiv \forall_n P_{(n)} \wedge \forall_n Q_{(n)}$$

$$\forall_n (P_{(n)} \vee Q_{(n)}) \stackrel{?}{=} \forall_n P_{(n)} \vee \forall_n Q_{(n)}$$

$$\forall_{\delta \in \text{Domain}} (P_{(\delta)} \vee Q_{(\delta)}) \xrightarrow{?} (\forall_{\delta \in \text{Domain}} P_{(\delta)}) \vee (\forall_{\delta \in \text{Domain}} Q_{(\delta)})$$

Domain:make $\alpha = \text{True}$

$$q = P_{(a)}^{\text{F}} \vee Q_{(a)}^{\text{T}} = \text{T}$$

$$b = P_{(b)}^{\text{T}} \vee Q_{(b)}^{\text{T}} = \text{T}$$

$$c = P_{(c)}^{\text{F}} \vee Q_{(c)}^{\text{T}} = \text{T}$$

;

β
Can we make $\beta = \text{False}$?

To make $\beta = \text{False}$
 $\forall_{\delta \in \text{Domain}} P_{(\delta)} = \text{false}$ ✓
 and
 $\forall_{\delta \in \text{Domain}} Q_{(\delta)} = \text{false}$ ✓

$$\forall_n (P(n) \vee Q(n)) \quad \times$$

$$\forall_n P(n) \vee \forall_n Q(n)$$

Domain: \mathbb{N}

$P(n) = n$ is even

$Q(n) = n$ is odd

$\alpha = \text{True}$

everyone is even
on

everyone is odd



$$\forall x (P(x) \vee Q(x))$$

Everyone is
even or odd

U

True

$$\forall x P(x) \vee \forall x Q(x)$$

Everyone is even

or
Everyone is odd

Everyone is odd = F

FVF = F

$$\underbrace{\forall \alpha (\rho_{(\alpha)} \vee \varphi_{(\alpha)})}_{\beta} \leftarrow \left(\underbrace{\forall \alpha \rho_{(\alpha)}}_{\varphi} \right) \vee \left(\underbrace{\forall \alpha \varphi_{(\alpha)}}_{\varphi} \right)$$

$$\underline{\beta = \text{True}}$$

To make $\alpha = \text{True}$

$$\underline{\forall \alpha \rho_{(\alpha)} = \text{True}} \leftarrow$$

or

$$\underline{\beta = \text{True}}$$

$$\underline{\forall \alpha \varphi_{(\alpha)} = \text{True}}$$



$$\forall_n (\rho_{(n)} \vee \varphi_{(n)}) \xrightleftharpoons[X]{} \forall_n \rho_{(n)} \vee \forall_n \varphi_{(n)}$$

GO
CLASSES



Statements involving predicates and quantifiers are *logically equivalent* if and only if they have the same truth value no matter which predicates are substituted into these statements and which domain of discourse is used for the variables in these propositional functions. We use the notation $S \equiv T$ to indicate that two statements S and T involving predicates and quantifiers are logically equivalent.





First Order Logic

Next Topic:

Distributive Properties of Quantifiers

(Distribution of Quantifiers over logical connectives)

Website : <https://www.goclasses.in/>



$$\forall x ((P(x) \wedge Q(x)) \xleftarrow{X} \forall x P(x) \wedge \forall x Q(x))$$

$$\forall x (P(x) \vee Q(x)) \xrightarrow{X} \forall x P(x) \vee \forall x Q(x)$$



Distribution of Quantifiers over logical
Connectives : $\Pi \rightarrow \underline{\text{Quantifier}}(\exists, \forall, \exists!)$

$$\Pi_x (P_{(x)} \# Q_{(x)}) \xrightarrow{?} \Pi_x P_{(x)} \# \Pi_x Q_{(x)}$$

$\wedge, \vee, \oplus, \rightarrow, \leftrightarrow, \uparrow, \downarrow$



$$\pi_n(p_{(n)} \# q_{(n)})$$

Compact Exp

$$\pi_n p_{(n)} \# \pi_n q_{(n)}$$

Expanded
Exp



\forall with \wedge ;

Compact $\xrightarrow{\checkmark}$ Expanded



\forall with \vee ,

Compact $\xleftarrow{\checkmark}$ Expanded



(\exists) with (\wedge):



$$\exists x (P(x) \wedge Q(x)) \rightarrow$$

there is someone for whom
P \wedge Q = True

logical thinking
method

$$\exists x P(x) \wedge \exists x Q(x)$$

There is someone for whom P = True,
AND

There is someone for whom Q = True

$$\exists x (P(x) \wedge Q(x)) \xrightarrow{\alpha} (\exists x P(x)) \wedge (\exists x Q(x)) \xrightarrow{\beta}$$

Domain:

Make $\alpha = \text{True}$

$$b \Rightarrow P(b) \wedge Q(b) = \text{True}$$

;

Can we make β false? \Rightarrow No

$$P(b) = \text{True}$$

$$\exists x P(x) = \text{True}$$

$$Q(b) = \text{True} \text{ so } \exists x Q(x) = \text{True}$$

$$\exists x(P(x) \wedge Q(x))$$

for someone $P \wedge Q = \text{True}$

↓
Cannot guarantee

$$\exists x P(x) \wedge \exists x Q(x)$$

for someone $M_P = \text{True}$

and

for someone $N_Q = \text{True}$

$$\exists x (P(x) \wedge Q(x))$$

false

None is

even prime AND odd

$$\exists x P(x) \wedge \exists x Q(x)$$

P(x) = even prime

Q(x) = odd

witness = 2

witness = 3

$$\exists x(P(x) \wedge Q(x)) \xleftarrow{\beta} \exists x P(x) \wedge \exists x Q(x)$$

make Q True

Can we make $\beta = \text{false}$

$$a \Rightarrow Q(a) = F$$

$$b \Rightarrow P(b) = F$$

$$c \Rightarrow P(c) = F$$

$$d \Rightarrow P(d) = F$$

$\beta = \text{false}$

Domain

a

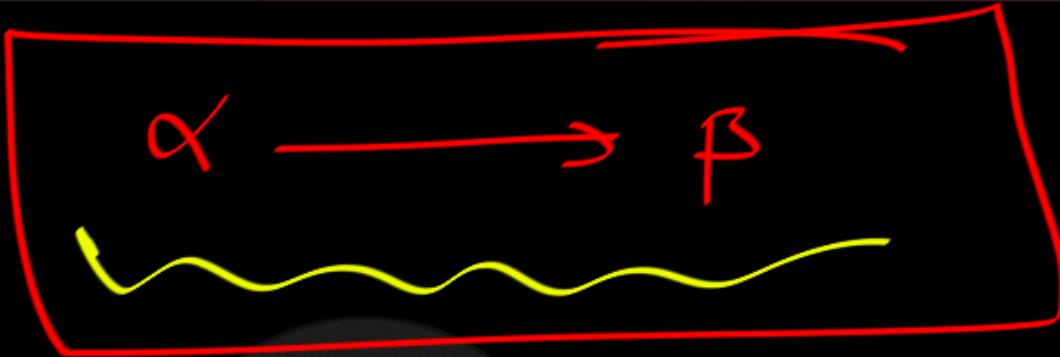
P = T ✓

b

c

Q = T

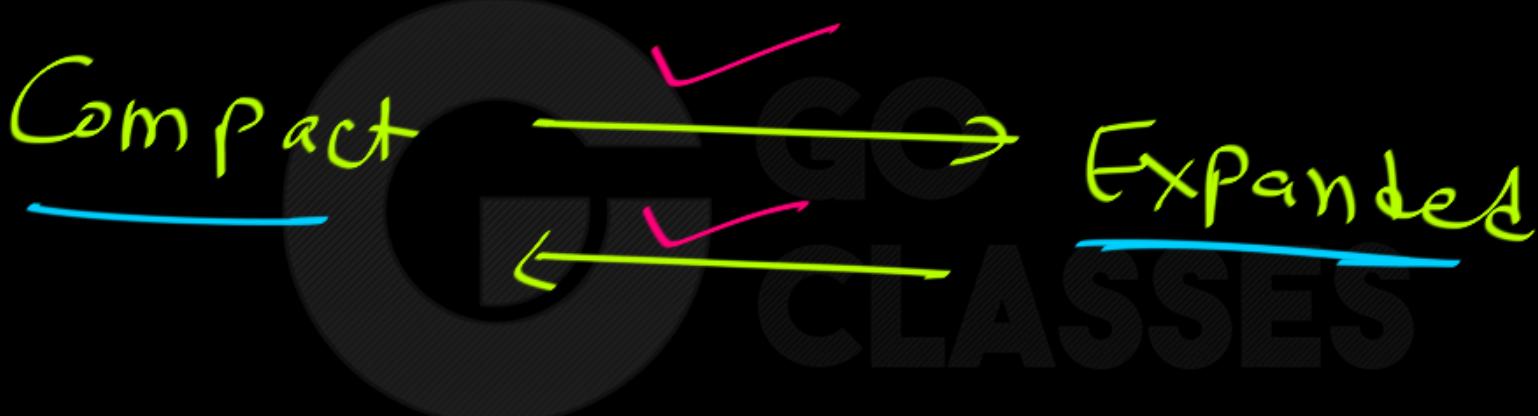
d



Domain: $\{a, b, c, d, \dots\}$



\exists with \forall :



$$\exists_{\forall n} (P_{(n)} \vee Q_{(n)}) \xrightarrow{\quad} \exists_{\forall n} P_{(n)} \vee \exists_{\forall n} Q_{(n)}$$

Someone m

$$\underline{P(m) \vee Q(m)} = \text{True}$$

P(m) or Q(m) is True.

Someone m

$$P = \text{True}$$

OR

Someone m

$$Q = \text{True}$$

$$\exists_{\forall n} (P_{(n)} \vee Q_{(n)}) \leftarrow \exists_{\forall n} P_{(n)} \vee \exists_{\forall n} Q_{(n)}$$

$\beta = \text{True}$ Someone M $P = \text{True}$

$\beta = \text{True}$ Someone N $Q = \text{True}$

~~OR~~

$$\exists_{\alpha} (P_{(\alpha)} \oplus Q_{(\alpha)}) \xrightarrow{\text{X}} \exists_{\alpha} P_{(\alpha)} \oplus \exists_{\alpha} Q_{(\alpha)}$$

α β

Domain:

a

b

c

make $\alpha = \text{True}$

$$(P_{(b)} = T) \oplus Q_{(b)} = \text{True}$$

Can we make β false?

$\exists_{\alpha} P_{(\alpha)} = \text{True}$ ($P_{(b)} = T$)

$\exists_{\alpha} Q_{(\alpha)} = \text{True}$

$Q_{(a)} = \text{True}$ ✓

$$\exists_{\exists \forall} (P_{(\forall)} \oplus Q_{(\forall)})$$

$$\exists_{\exists \forall} P_{(\forall)} \oplus \exists_{\exists \forall} Q_{(\forall)}$$

β

Can make $\beta = \text{false}$

Domain: $\{q\}$ $P_{(q)} = T, Q_{(q)} = F$

b

c

d

witness
for β

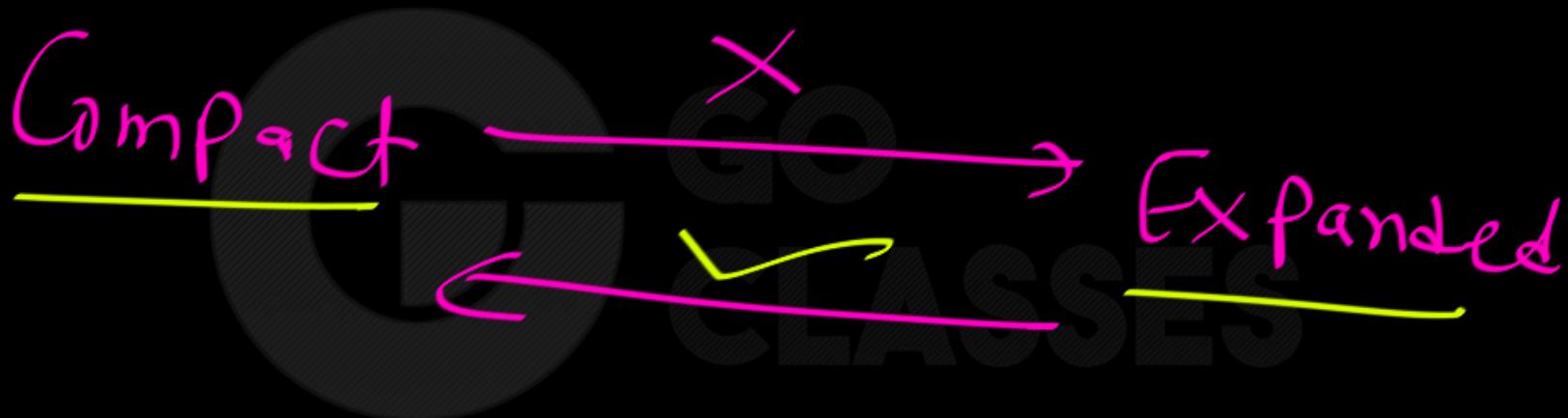
make $\alpha = \text{True}$
 $\exists_{\forall} P_{(\forall)} = \text{True}$ $P_{(q)} = T$
AND

$\exists_{\forall} Q_{(\forall)} = \text{False}$

for everyone $Q = \text{false}$



\exists with \oplus ;



→ means
← means

Compact → Expanded
Compact ← Expanded

28
Question

10 Done

Hw:
18 Q.

	+	?
^	↔ (proven) ↔ (proven)	→ (proven) ↔ (proven)
→	→	←
↔	→	×
⊕	×	↔ (proven)
↑	→	↔
↓	→	↔



Q
msd

- Ⓐ $\forall (\wedge) \equiv \forall \wedge \wedge \forall$
- Ⓑ $\exists (\vee) \equiv \exists \vee \vee \exists$
- Ⓒ $\forall (\rightarrow) \equiv \forall \rightarrow \rightarrow \forall$
- Ⓓ $\forall (\oplus) \equiv \forall \oplus \oplus \forall$
- 8 Questions

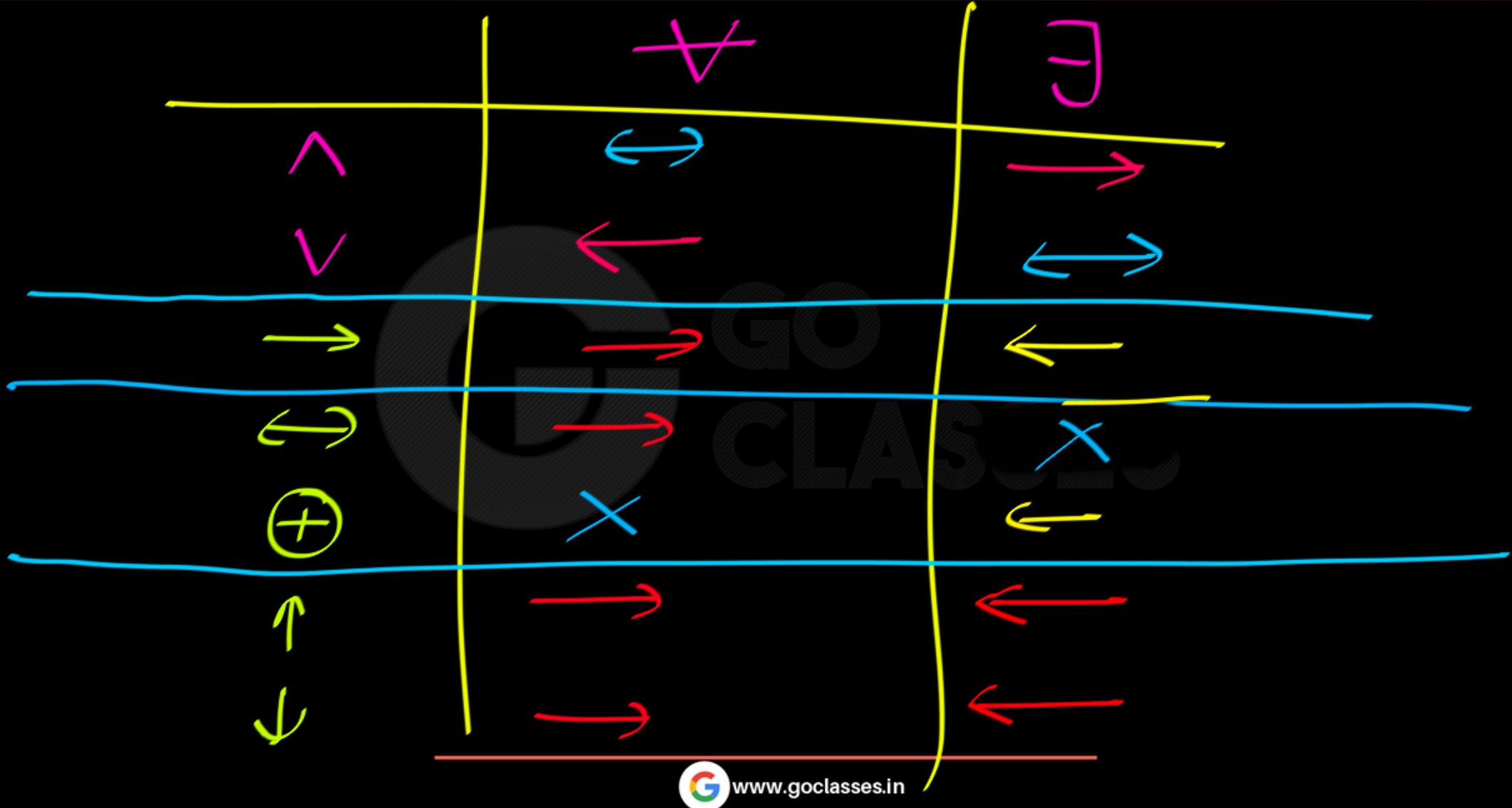
Note: Don't ^{blindly} by-heart this table.

Prove Complete Table then

Remember this Table.



Discrete Mathematics



Q: Which is valid?

① $\left[\forall_{\alpha} P_{(\alpha)} \rightarrow \forall_{\alpha} Q_{(\alpha)} \right] \rightarrow \forall_{\alpha} (P_{(\alpha)} \rightarrow Q_{(\alpha)})$

② $\exists_{\alpha} (P_{(\alpha)} \uparrow Q_{(\alpha)}) \rightarrow \exists_{\alpha} P_{(\alpha)} \uparrow \exists_{\alpha} Q_{(\alpha)}$

③ $\exists_{\alpha} (P_{(\alpha)} \leftrightarrow Q_{(\alpha)}) \rightarrow \exists_{\alpha} P_{(\alpha)} \leftrightarrow \exists_{\alpha} Q_{(\alpha)}$

Q: Which is valid?

~~1~~ $\forall_{\forall x} P(x) \rightarrow \forall_x Q(x)$ $\xrightarrow{x} \forall_{\forall x} (P(x) \rightarrow Q(x))$ ✓

~~2~~ $\exists_x (P(x) \uparrow Q(x)) \xrightarrow{x} \exists_x P(x) \uparrow \exists_x Q(x)$ ✓

~~3~~ $\exists_x (P(x) \leftrightarrow Q(x)) \xrightarrow{x} \exists_x P(x) \leftrightarrow \exists_x Q(x)$ ✗



Q: Which is valid?

$$\textcircled{1} \quad \left[\forall_{\forall x} P_{(x)} \oplus \forall_{\forall x} Q_{(x)} \right] \rightarrow \forall_{\forall x} (P_{(x)} \oplus Q_{(x)})$$

$$\textcircled{2} \quad \exists_x (P_{(x)} \downarrow Q_{(x)}) \leftarrow \exists_x P_{(x)} \downarrow \exists_x Q_{(x)}$$

$$\textcircled{3} \quad \forall_x (P_{(x)} \leftrightarrow Q_{(x)}) \rightarrow \forall_x P_{(x)} \leftrightarrow \forall_x Q_{(x)}$$

Q: Which is valid?

~~1~~ $\left[\forall_{\forall x} P_{(x)} \oplus \forall_{\forall x} Q_{(x)} \right] \xrightarrow{x} \forall_{\forall x} (P_{(x)} \oplus Q_{(x)})$

2 $\exists_x (P_{(x)} \downarrow Q_{(x)}) \xleftrightarrow{x} \exists_x P_{(x)} \downarrow \exists_x Q_{(x)}$

~~3~~ $\forall_x (P_{(x)} \leftrightarrow Q_{(x)}) \xrightarrow{x} \forall_x P_{(x)} \leftrightarrow \forall_x Q_{(x)}$

$\varphi:$

$$\forall_{\alpha} (P_{(\alpha)} \leftrightarrow Q_{(\alpha)})$$

 φ

Invalid

make $\alpha = \text{True}$

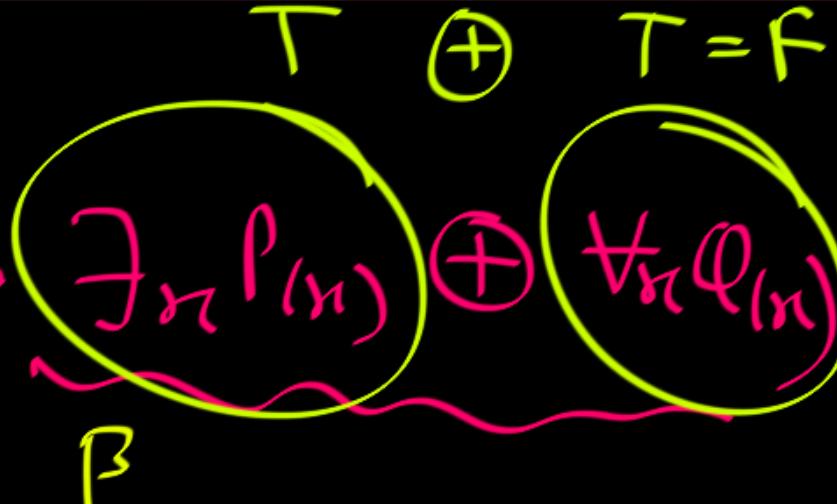
$$a - P = Q$$

$$b - P = Q$$

$$c - P = Q$$

?

?

Can we make $\beta = \text{false}$?

$$\left. \begin{array}{l} a - P = Q = T \\ b - P = Q = T \\ c - P = Q = T \end{array} \right\} \Rightarrow \begin{array}{l} P_{(a)} = T \\ P_{(b)} = T \\ P_{(c)} = T \end{array}$$



Q:

$$\forall_{\delta\in\mathbb{N}} \left(P_{(\delta)} \leftrightarrow Q_{(\delta)} \right) \leftarrow \exists_{\delta\in\mathbb{N}} P_{(\delta)} \oplus \forall_{\delta\in\mathbb{N}} Q_{(\delta)}$$



$\varphi:$

$$\forall_{\alpha} \left(P_{(\alpha)} \leftrightarrow Q_{(\alpha)} \right)$$

β

make $\beta = \text{false}$

$$q \rightarrow Q_{(q)} = F$$

①: $P_{(d)} = T$

$$b \rightarrow Q_{(b)} = F$$

$$c \rightarrow Q_{(c)} = F$$

Invalid

$$\exists_{\alpha} P_{(\alpha)} \oplus \forall_{\alpha} Q_{(\alpha)}$$

φ

Can we make $\alpha = \text{True}$

$$\exists_{\alpha} P_{(\alpha)} = \text{True} (P_{(d)} = T)$$

$\forall_{\alpha} Q_{(\alpha)} = \text{false} \Rightarrow$ for everyone $\varphi = \text{False}$



Which of the following semantic entailments are valid in predicate logic?

1. $\forall x (P(x) \vee Q(x)) \models \forall x P(x) \vee \forall x Q(x)$
2. $\forall x (P(x) \rightarrow Q(x)) \models \forall x P(x) \rightarrow \forall x Q(x)$
3. $\forall x P(x) \rightarrow \forall x Q(x) \models \forall x (P(x) \rightarrow Q(x))$
4. $\neg \forall x (P(x) \wedge Q(x)) \models \exists x \neg P(x) \wedge \exists x \neg Q(x)$
5. $\exists x P(x) \wedge \exists x Q(x) \models \exists x (P(x) \wedge Q(x))$