# Answers to Some Predicate Logic Formalisation Exercises from the Predicate Logic Notes

Predicates to be used:

lecTheatre/1                              same/2

office/1                                  phd/1

contains/2                                supervises/2

lecturer/1                                happy/1

has/2                                     completePhd/1

1.

311 is a lecture theatre and 447 is an office.

311 is a lecture theatre and 447 is an office

∧

lecTheatre(311)   ∧   office(447)

2. Every lecture theatre contains a projector.

Every lecture theatre contains a projector.

$\forall \mathbf{x}$ ( Assignment → Project ) Exam Help

https://powcoder.com

Add WeChat powcoder

Every

Any

All

Everyone

….

Every lecture theatre contains a projector.

$\forall$X (lecTheatre(X) $\rightarrow$          )

$\forall$X (lecTheatre(X) $\rightarrow$ contains(X, projector))

Alternatively:

$\forall$X (lecTheatre(X) $\rightarrow$

$\exists$Y (projector(Y) $\wedge$ contains(X, Y)))

A universally quantified wff is usually like this:

∀x (        →        )

Or like this:

∀x (        ↔        )

That is their principle connective is

→ or ↔.

3. Every office contains a telephone and either a desktop or a laptop computer.

Every office contains a telephone and either a desktop or a laptop computer.

∀X (office(X) → …… )

… contains a telephone and either a desktop or a laptop computer.

∀X (office(X) →( …. ∧ …))

∀X (office(X) →(contains(X, telephone) ∧ (…)))

… either a desktop or a laptop computer.

∀X (office(X) →(contains(X, telephone) ∧ ( … ∨ …)))

∀X (office(X) →(contains(X, telephone) ∧

(contains(X, desktop) ∨ contains(X, laptop))))

4. Every lecturer has at least one office.

Every lecturer has at least one office.

∀X (lecturer(X) → ... )

... has at least one office.

∀X (lecturer(X) → ∃Y (office(Y) ∧ has(X, Y)))

At least one

Some

One

....

map to ∃

An existentially quantified wff is usually like this:

∃**x** (          ∧          )

Or like this:

∃**x** (          ∨          )

That is their principle connective is

∧ or ∨.

5. No lecturer has more than one office.

No lecturer has more than one office.

Not any lecturer has more than one office.

(There does not exist a lecturer who has more than one office.)

¬∃L (lecturer(L) ∧ *L has more than one office*)

¬∃L (lecturer(L) ∧ *L has at least 2 offices that are not the same*)

¬∃L (lecturer(L) ∧ there are *at least* *2 offices that L has and are not the same*)

¬∃L (lecturer(L) ∧

∃O1 ∃O2 (office(O1) ∧ office(O2) ∧

has(L, O1) ∧ has(L, O2) ∧

¬same(O1,O2)))

There are other ways of doing 5. For example:

$\forall$L (lecturer(L) → ¬ ($\exists$O1 $\exists$O2 (office(O1) $\land$ office(O2) $\land$ has(L, O1) $\land$ has(L, O2) $\land$ ¬same(O1,O2))))

$\forall$L $\forall$O1 $\forall$O2 (lecturer(L) $\land$ office(O1) $\land$ office(O2) $\land$ has(L, O1) $\land$ has(L, O2) → same(O1,O2))

6. No lecturers share offices with anyone.

Try it .....

7. Some lecturers supervise PhD students and some do not.

$(\exists L \ .....) \wedge (\exists L .....)$ or

$(\exists L1 \ .....) \wedge (\exists L2 .....)$

Some lecturers supervise PhD students and some do not.

$(\exists L\ (lecturer(L) \wedge$ *there is at least one PhD student that L supervises*$)) \wedge$

$(\exists L\ (lecturer(L) \wedge$ *there is not at least one PhD student that L supervises*$))$

$(\exists L\ (lecturer(L) \wedge \exists S\ (phd(S) \wedge supervises(L,S)))) \wedge$

$(\exists L\ (lecturer(L) \wedge \neg \exists S\ (phd(S) \wedge supervises(L,S))))$

($\exists$L (lecturer(L) $\wedge$ $\exists$S (phd(S)$\wedge$ supervises(L,S))))$\wedge$

($\exists$L (lecturer(L) $\wedge$ $\neg\exists$S (phd(S) $\wedge$ supervises(L,S))))

Drop some unneccesary brackets:

$\exists$L (lecturer(L) $\wedge$ $\exists$S (phd(S)$\wedge$ supervises(L,S)))$\wedge$

$\exists$L (lecturer(L) $\wedge$ $\neg\exists$S (phd(S) $\wedge$ supervises(L,S)))

8. Each PhD student has an office, but all PhD students share their office with at least one other PhD student.

Try it ......

9. A lecturer is happy if the PhD students he/she supervises successfully complete their PhD.

∀L (lecturer(L) ∧ *his/her PhD students successfully complete* → happy(L))

∀L (lecturer(L) ∧ *all his/her PhD students successfully complete* → happy(L))

∀L (lecturer(L) ∧ ∀S *if S is L's PhD student then S successfully completes* → happy(L))

∀L (lecturer(L) ∧

∀S *if S is L's PhD student then S successfully completes*→ happy(L))

∀L (lecturer(L) ∧

∀S (phd(S) ∧ supervises(L,S) → completePhD(S))

→ happy(L))

10.Not all PhD students complete their PhD.

Try it ....

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder