

AI

LAB 6

Propositional Logic

Implementation of truth-table enumeration algorithm for deciding propositional entailment. i.e., Create a knowledge base using propositional logic and show that the given query entails the knowledge base or not.

```
from itertools import product

# Define propositional logic operations
def implies(a, b):
    return (not a) or b

# Knowledge Base sentences
def KB(P, Q, R):
    s1 = implies(Q, P) #  $Q \rightarrow P$ 
    s2 = implies(P, not Q) #  $P \rightarrow \neg Q$ 
    s3 = Q or R #  $Q \vee R$ 
    return s1 and s2 and s3 # KB is true only if all hold

# All combinations of truth values for P, Q, R
values = list(product([False, True], repeat=3))

print("P\tQ\tR\tQ→P\tP→¬Q\tQVR\tKB")
print("-"*50)

models = []
for P, Q, R in values:
    s1 = implies(Q, P)
    s2 = implies(P, not Q)
    s3 = Q or R
    kb_val = s1 and s2 and s3
    print(f"{P}\t{Q}\t{R}\t{s1}\t{s2}\t{s3}\t{kb_val}")
    if kb_val:
        models.append((P, Q, R))

print("\n Models where KB is True:", models)

# Check entailments
entails_R = all(R for P, Q, R in models)
entails_R_imp_P = all((not R) or P for P, Q, R in models)
entails_Q_imp_R = all((not Q) or R for P, Q, R in models)
```

```

print("\nEntailments:")
print("KB  $\models$  R :", entails_R)
print("KB  $\models$  R  $\rightarrow$  P :", entails_R_imp_P)
print("KB  $\models$  Q  $\rightarrow$  R :", entails_Q_imp_R)

```

OUTPUT:

\Rightarrow	P	Q	R	$Q \rightarrow P$	$P \rightarrow \neg Q$	$Q \vee R$	KB
	False	False	False	True	True	False	False
	False	False	True	True	True	True	True
	False	True	False	False	True	True	False
	False	True	True	False	True	True	False
	True	False	False	True	True	False	False
	True	False	True	True	True	True	True
	True	True	False	True	False	True	False
	True	True	True	True	False	True	False

Models where KB is True: [(False, False, True), (True, False, True)]

Entailments:
 KB \models R : True
 KB \models R \rightarrow P : False
 KB \models Q \rightarrow R : True