Lab 3

May 31, 2024

Dylan Liesenfelt

1 Python and Compound Statements

2

2.1 Logic Statement in LaTeX

$$(p \rightarrow q) \land (\neg r \lor (q \leftrightarrow p))$$

2.1.1 Python Logic Statement

[ ]:

```python
# Variables
p = True
q = False
r = True

# Logic statement
logic_statement = (not p or q) and (not r or (q == p))

# Print the result
print(f"For p = {p}, q = {q}, r = {r}, the result is {logic_statement}")

# Test with different values
test_cases = [
    (True, True, True),
    (True, True, False),
    (True, False, True),
    (True, False, False),
    (False, True, True),
    (False, True, False),
    (False, False, True),
    (False, False, False)
]

for p, q, r in test_cases:
    result = (not p or q) and (not r or (q == p))
    print(f"For p = {p}, q = {q}, r = {r}, the result is {result}")
```

```
For p = True, q = False, r = True, the result is False
For p = True, q = True, r = True, the result is True
For p = True, q = True, r = False, the result is True
For p = True, q = False, r = True, the result is False
For p = True, q = False, r = False, the result is False
For p = False, q = True, r = True, the result is False
For p = False, q = True, r = False, the result is True
For p = False, q = False, r = True, the result is True
For p = False, q = False, r = False, the result is True
```

[ ]:

```python
# Define implies function
def implies(p, q):
    return not p or q

# Columns
col1 = 'p'
col2 = 'q'
col3 = '¬p'
col4 = '¬p   q'
col5 = 'p → q'
print(f'{col1} \t{col2} \t{col3} \t{col4} \t{col5}')
print('-'*40)

# Iteration of proposition values
for p in [True, False]:
    for q in [True, False]:
        not_p = not p
        not_p_or_q = not_p or q
        p_implies_q = implies(p, q)
        print(f'{p} \t{q} \t{not_p} \t{not_p_or_q} \t{p_implies_q}')
```

```
p       q       ¬p      ¬p   q  p → q
----------------------------------------
True    True    False   True    True
True    False   False   False   False
False   True    True    True    True
False   False   True    True    True
```