Assignment 4

February 22, 2020

1 Assignment 4

1.1 Problem 1

Create a function fa(p,q,r). It returns the value of the logical expression

$$p \wedge (q \vee r)$$

Use this function to print a complete truth table for the logical expression.

[13]:
```
TF= [True, False]
def fa(p, q, r):
    return p and (q or r)

for p in TF:
    for q in TF:
        for r in TF:
                print("p:", p, "| q:", q, "| r:", r)
                print("              qVr:", (q or r))
                print("    p/\(qVr):", fa(p, q, r))
                print()
```

```
p: True | q: True | r: True
              qVr: True
    p/\(qVr): True

p: True | q: True | r: False
              qVr: True
    p/\(qVr): True

p: True | q: False | r: True
              qVr: True
    p/\(qVr): True

p: True | q: False | r: False
              qVr: False
    p/\(qVr): False
```

```
p: False | q: True | r: True
              qVr: True
    p/\(qVr): False

p: False | q: True | r: False
              qVr: True
    p/\(qVr): False

p: False | q: False | r: True
              qVr: True
    p/\(qVr): False

p: False | q: False | r: False
              qVr: False
    p/\(qVr): False
```

1.2 Problem 2

Create a function fb(p,q,r). It returns the value of the logical expression

$$(p \wedge q) \vee (p \wedge r))$$

Use this function to print a complete truth table for the logical expression.

[16]:
```
TF= [True, False]
def fb(p, q, r):
    return (p and q) or (p and r)

for p in TF:
    for q in TF:
        for r in TF:
                print("p:", p, "| q:", q, "| r:", r)
                print("   p /\ q:", p and q, "| p /\ r:", (p and r))
                print("    (p /\ q)V(p /\ r):", fa(p, q, r))
                print()
```

```
p: True | q: True | r: True
   p /\ q: True | p /\ r: True
    (p /\ q)V(p /\ r): True

p: True | q: True | r: False
   p /\ q: True | p /\ r: False
    (p /\ q)V(p /\ r): True

p: True | q: False | r: True
   p /\ q: False | p /\ r: True
```

```
    (p /\ q)V(p /\ r): True

p: True | q: False | r: False
   p /\ q: False | p /\ r: False
    (p /\ q)V(p /\ r): False

p: False | q: True | r: True
   p /\ q: False | p /\ r: False
    (p /\ q)V(p /\ r): False

p: False | q: True | r: False
   p /\ q: False | p /\ r: False
    (p /\ q)V(p /\ r): False

p: False | q: False | r: True
   p /\ q: False | p /\ r: False
    (p /\ q)V(p /\ r): False

p: False | q: False | r: False
   p /\ q: False | p /\ r: False
    (p /\ q)V(p /\ r): False
```

1.3 Problem 3

Write a python program which uses the two functions you created to answer a question. Are these two logical expressions equivalent? You should modify the function compare(), which I demonstrated in class, to handle three logical

[17]:
```python
def compare(fa, fb):
    TF = [True, False]
    fal = []
    fbl = []

    for p in TF:
        for q in TF:
            for r in TF:
                a = fa(p, q, r)
                b = fb(p, q, r)
                fal.append(a)
                fbl.append(b)
    if a == b:
        conclusion = "Yes - Equivalent"
    else:
        conclusion = "No - Not Equivalent"
    return conclusion
```

```
compare(fa, fb)
```

[17]: 'Yes - Equivalent'

[0]: