NAME-- RAJNISH KUMAR NO. -- 6846168

1. Please give the logical expression of the logical circuit in the box below (Task 1).

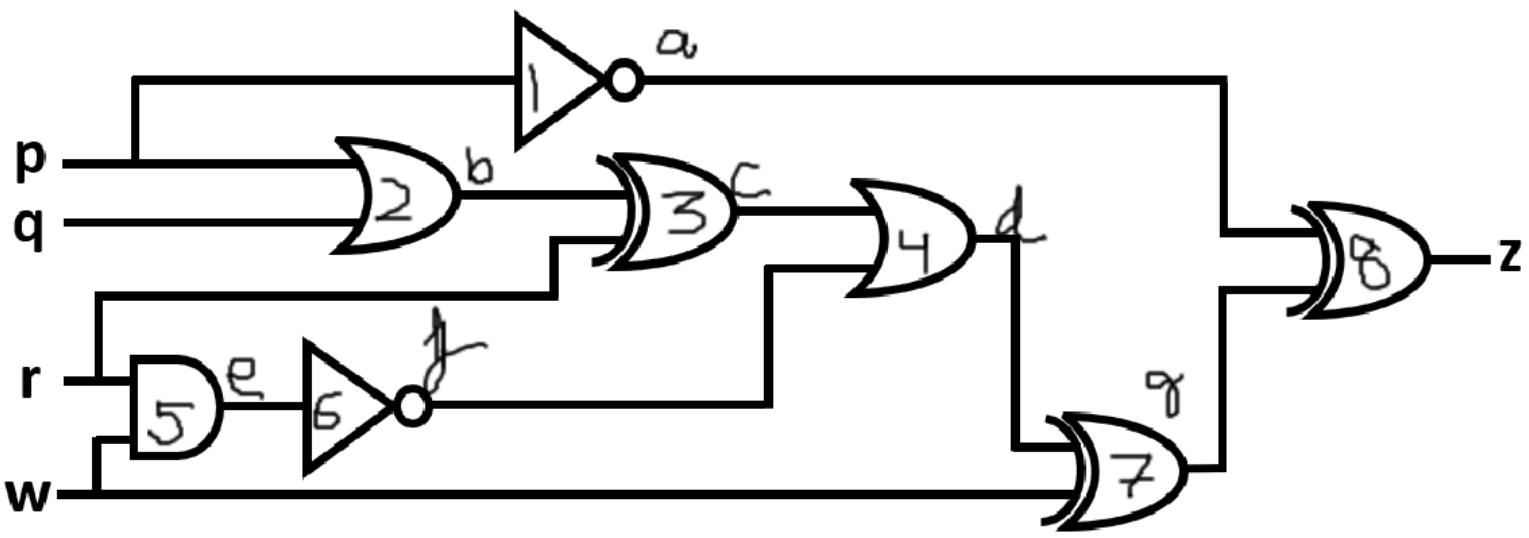

Fig. 1. The logical circuit

∧ (and), ⊕ (Xor), ∨ (or) and ¬ (not)

#--To derive logical expression I have expanded output and move towards input. For this I need to assign logical gates with numbers (1, 2, 3, 4, 5, 6, 7, 8) and their outputs with letters (a, b, c, d, e, f, g, z) see in diagram.

#--lets start with output first which is Xor gate (8) with output z and it has two input a and g

z= a Xor g
z= a ⊕ g

#--Expanding a and g. a is the output of Not gate(1) with p as input and g is the output of Xor gate(7) with two inputs # d and w.

z= (¬p) ⊕ (d Xor w)
z= (¬p) ⊕ (d ⊕ w)

#--Expainding d, Or gate(4) with two inputs c and f

z= (¬p) ⊕ ((c ∨ f) ⊕ w)

#--Expanding f, Not gate(6) with input e

z= (¬p) ⊕ ((c ∨ (¬e)) ⊕ w)

#--Expanding e, And gate(5) with input r and w

z= (¬p) ⊕ ((c ∨ (¬(r ∧ w))) ⊕ w)

#Exapnding c, Xor gate(3) with inputs b and r

z= (¬p) ⊕ (((b ⊕ r) ∨ (¬(r ∧ w))) ⊕ w)

#--Expanding b, And gate (2) with inputs p and q.

z= (¬p) ⊕ ((((p ∨ q) ⊕ r) ∨ (¬(r ∧ w))) ⊕ w)

2. Please complete the truth table of the logical circuit given by the text box below (Task 1).

Please provide the missing values in the z column to complete the truth table.

Truth table of the logical circuit

p	q	r	w	z
1	1	1	1	1
1	1	1	0	1

p	q	r	w	z
1	1	0	1	0
1	1	0	0	1
1	0	1	1	1
1	0	1	0	1
1	0	0	1	0
1	0	0	0	1
0	1	1	1	0
0	1	1	0	0
0	1	0	1	1
0	1	0	0	0
0	0	1	1	1
0	0	1	0	0
0	0	0	1	1
0	0	0	0	0

3. Please complete the code template of the logical circuit given by the code box below (Tasks 2 & 3).

In [1]:
```
def logical_circuit(p, q, r, w):

    # Using if statement so that only integer values can be inserted
    if not (isinstance(p, int) and isinstance(q, int) and isinstance(r, int) and isinstance(w, int)):
        raise ValueError("Inputs must be integers")
    # Using if statement so that only 0 and 1 can act as input.
    if p not in (0, 1) or q not in (0, 1) or r not in (0, 1) or q not in (0, 1):
        raise ValueError("Inputs must be 0 or 1.")




    # Defining basic Logic gates.
    def Not(x):    return not x
    def And(x,y):  return x and y
    def Or(x, y):  return x or y
    def Xor(x, y):
        if (x!=y): return 1
        elif (x==y):return 0


    # Logical circuit For this I need to assign logical gates with numbers (1, 2, 3, 4, 5, 6, 7, 8)
    # and their outputs with letters (a, b, c, d, e, f, g, z)-- See in diagram. example g7= Xor(d4, w) is a Xor gate
    # represented by no. 7 in diagram having two inputs d4 and w with output g.
    a1= Not(p)
    b2= Or(p,q)
    c3= Xor(b2, r)
    e5= And(r,w)
    f6= Not(e5)
    d4= Or(c3, f6)
    g7= Xor(d4,w)
    z = Xor(a1, g7)


    return z
```

In [2]:
```
print(logical_circuit(1,1,1,1))
print(logical_circuit(1,1,1,0))
print(logical_circuit(1,1,0,1))
print(logical_circuit(1,1,0,0))
print(logical_circuit(1,0,1,1))
print(logical_circuit(1,0,1,0))
print(logical_circuit(1,0,0,1))
print(logical_circuit(1,0,0,0))
print(logical_circuit(0,1,1,1))
print(logical_circuit(0,1,1,0))
print(logical_circuit(0,1,0,1))
print(logical_circuit(0,1,0,0))
print(logical_circuit(0,0,1,1))
print(logical_circuit(0,0,1,0))
print(logical_circuit(0,0,0,1))
print(logical_circuit(0,0,0,0))
p= int(input("Enter p:"))
q= int(input('Enter q:'))
r= int(input('Enter r:'))
w= int(input('Enter w:'))
logical_circuit(p, q, r, w)
```

```
1
1
0
1
0
0
1
0
1
0
1
0
Enter p:1
Enter q:1
Enter r:1
Enter w:1
```

Out[2]: 1

References-- 1-- https://learn.sparkfun.com/tutorials/logicblocks--digital-logic-introduction/all (https://learn.sparkfun.com/tutorials/logicblocks--digital-logic-introduction/all)

2--https://www.geeksforgeeks.org/logic-gates-in-python/ (https://www.geeksforgeeks.org/logic-gates-in-python/)

3--https://docs.python.org/3/library/functions.html#isinstance (https://docs.python.org/3/library/functions.html#isinstance)

In [ ]: