Tutorial 5 Week of 31 March, 2008

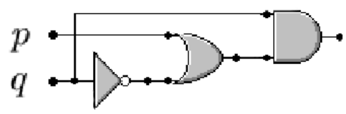
Logical gates and expressions

(i) Write pseudo-code for the logical function AND ($\wedge$), OR ($\vee$), XOR ($\oplus$) and NOT ($\neg$) using the arithmetic operators and a function MAX which, applied to a list, returns the largest element of the list.

```
[0] r = x AND y
   [0.1] NB. x and y are Booleans
   [0.1] NB. r is a Boolean and 1 only when x and y are both 1
[1] r ← x ⋆ y

[0] r = x XOR y
   [0.1] NB. x and y are Booleans
   [0.1] NB. r is a Boolean and 1 when either x and y is 1 but not both
[1] r ← (x > y) + (x < y)

[0] r = x OR y
   [0.1] NB. x and y are Booleans
   [0.1] NB. r is a Boolean and 1 when either x and y is 1
[1] r ← (x + y) − x ⋆ y

[0] r = NOT y
   [0.1] NB. y is a Booleans
   [0.1] NB. r is a Boolean and 1 when y is 0
[1] r ← 1 − y
```

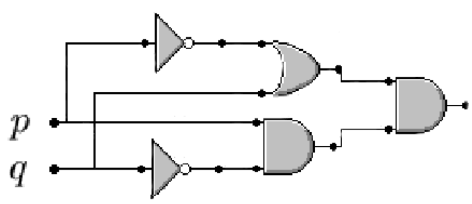
Write algebraic statements for each of the following circuit diagrams.

(ii)



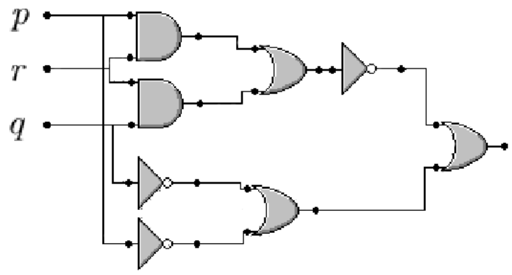
$p \wedge (p \vee \neg q)$

(iii)



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

(iv)

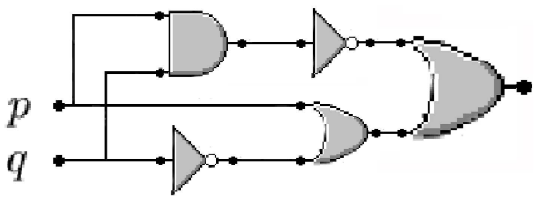


$$\neg((p \wedge r) \vee (q \wedge r)) \vee (\neg p \vee \neg q)$$

Draw circuit diagrams for each of the following expressions of classical logic.

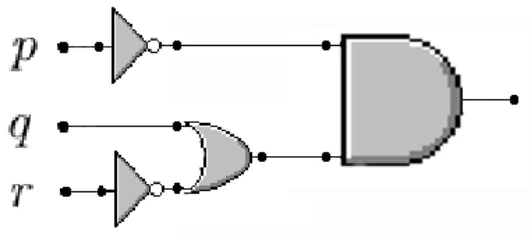
(v)

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



(vi)

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



(vii)

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