2.4) 20, 21, 28, 33
2.5) 33-36
Proof) NAND is functionally complete

For each of the tables in 18-21, construct (a) a Boolean expression having the given table as its truth table and (b) a circuit having the given table as its input/output table

20)

p	q	r	s	
1	1	1	1	p q r
1	1	0	0	
1	0	1	1	p ~q r
1	0	0	0	
0	1	1	0	
0	1	0	0	
0	0	1	0	
0	0	0	1	p ~q ~r

(p q r) (p ~q r) (p ~q ~r)

21)

p	q	r	s	
1	1	1	0	
1	1	0	1	p q ~r
1	0	1	0	
1	0	0	0	
0	1	1	1	~p q r
0	1	0	1	~p q ~r
0	0	1	0	
0	0	0	0	

(p q ~r) (~p q r) (~p q ~r)

Use the properties listed in Theorem 2.1.1 to show that each pair of circuits in 26-29 have the same input/output table. (Find the Boolean expressions for the circuits and show that they are logically equivalent when regarded as statement forms.)

```
28.
(P   Q)   (P   ~Q)   (~P   ~Q)   (P   Q)   (~P   ~Q)   (P   ~Q)
                                    (P   ~Q)

33.
a. Show that for the Sheffer stroke |,
        P   Q   (P | Q) | (P | Q).

P | Q   (P   Q)

(P | Q) | (P | Q)   ~((P | Q)   (P | Q))      by definition of |
                    ~(~(P   Q)   ~(P   Q))    by definition of |
                    ~((~P   ~Q)   (~P   ~Q))  by de morgan's law
                    ~(~P   ~Q)   ~(~P   ~Q)   by de morgan's law
                    (~~P   ~~Q)   (~~P   ~~Q) by de morgan's law
                    (P   Q)   (P   Q)         by double negative law
                    P   Q                     by idempotent laws



b. Use the results of Example 2.4.7 and part (a) above to
write P   ( Q   R) using only Sheffer strokes.


P   ( Q   R)   ~(~(P   ( Q   R)))   by double negative law
               ~(~P   ~( Q   R))    by de moregan's laws
               ~(~P   (~ Q   ~R))   by de moregan's laws
               ~(~P   (Q   ~R))     by double negative law
               ~~P   ~(Q   ~R)      by de morgan's laws
               ~~P   (Q | ~R)       by definition of |
               P   (Q | ~R)         by by double negative law
               ~(~(P   (Q | ~R)))   by double negative law
               ~(P | (Q | ~R))      by by definition of |


2.5
Use 8-bit representations to compute the sums in 31-36.
33. (-6) + (-73)
-6 in binary = 2^8 - |-6|
             = 256 - 6
             = 250
         = (1*2^7) + (1*2^6) + (1*2^5) + (1*2^4) + (1*2^3) + (0*2^2) + (1*2^1) + (0*2^0)
             = 11111010
```

```
-73 in binary = 2^8 - |-73|
              = 256 - 73
              = 250
         = (1*2^7) + (0*2^6) + (1*2^5) + (1*2^4) + (0*2^3) + (1*2^2) + (1*2^1) + (1*2^0)
              = 10110111

  11111010
+ 10110111
-----------
 110110001


34. 89 + (-55)

89 in binary = (0*2^7) + (1*2^6) + (0*2^5) + (1*2^4) + (1*2^3) + (0*2^2) + (0*2^1) + (1*2^0)
             = 01011001

-55 in binary = 2^8 - |-55|
              = 256 - 55
              = 250
         = (1*2^7) + (1*2^6) + (0*2^5) + (0*2^4) + (1*2^3) + (0*2^2) + (0*2^1) + (1*2^0)
              = 11001001

  01011001
+ 11001001
-----------
 100100010


35. (-15) + (-46)

-15 in binary = 2^8 - |-15|
              = 256 - 15
              = 250
         = (1*2^7) + (1*2^6) + (1*2^5) + (1*2^4) + (0*2^3) + (0*2^2) + (0*2^1) + (1*2^0)
              = 11110001

-46 in binary = 2^8 - |-46|
              = 256 - 46
              = 250
         = (1*2^7) + (1*2^6) + (0*2^5) + (1*2^4) + (0*2^3) + (0*2^2) + (1*2^1) + (0*2^0)
              = 11010010

  11110001
+ 11010010
-----------
```

```
 111000011
```

36. 123 + (-94)

```
123 in binary = (0*2^7) + (1*2^6) + (1*2^5) + (1*2^4) + (1*2^3) + (0*2^2) + (1*2^1) + (1*2^0)
              = 01111011

-94 in binary = 2^8 - |-94|
              = 256 - 94
              = 250
         = (1*2^7) + (0*2^6) + (1*2^5) + (0*2^4) + (0*2^3) + (0*2^2) + (1*2^1) + (0*2^0)
              = 10100010

  01111011
+ 10100010
-----------
 100011101
```

So here is what is needed to prove:

Pre-req

1. What is NAND?

NAND is the logical equivalent of AND + NOT

```
|  IN   | OUT  |
|:-----:|:----:|
| p | q | p|q  |
|:-:|:-:|:----:|
| 1 | 1 |  1   |
| 1 | 1 |  0   |
| 1 | 0 |  1   |
| 1 | 0 |  0   |
```

2. What is functionally complete? What certifies it?

Attempt:

Knowing the definition of functionally complete and what is needed to certify it, and what NAND i