Name: Julia Nelson Date: 02/14/2020

Pledge: " I pledge my honor that I have abided by the Stevens Honor System."

For each function below, trace through it with reasonably small integer values. What does each function do?
HINT: You should assume integers are 8 bits for the purpose of this exercise.

```
int mystery1(int a, int b) {
    int c = a - b,
        d = (c >> 7) & 1,
        mystery = a - c * d;
    return mystery;
}
```

Trace: mystery1(3, 7) returns **7 …(binary 00000111)**
Trace: mystery1(8, 7) returns **8 …(binary 00001000)**

Summary: **Mystery1 returns the max of two integer bits**

```
void mystery2(int values[], int i, int j) {
    values[i] = values[i] ^ values[j];
    values[j] = values[i] ^ values[j];
    values[i] = values[i] ^ values[j];
}
```

Note: Improper C++ syntax found below.
Trace: mystery2([1, 2, 3, 4], 0, 3) values = [**4**, **2**, **3**, **1**]
Trace: mystery2([1, 2, 3, 4], 1, 2) values = [**1**, **3**, **2**, **4**]

Summary: **Mystery2 swaps the values of indices I and j then returns**

```
int mystery3(int x, int y) {
    int s, c;
    s = x ^ y;
    c = x & y;
    while (c != 0) {
        c = c << 1;
        x = s;
        y = c;
        s = x ^ y;
        c = x & y;
    }
    return s;
}
```

Trace: mystery3(5, 7) returns **12 …(binary 00001100)**
Trace: mystery3(2, 8) returns **10 …(binary 00001010)**

Summary: **Mystery3 returns the sum of two integer inputs**

Lab 4 Work
Bitwise + Bitshift Operators

Julia Nelson

"I pledge my honor that I have abided by the Stevens Honor System."

① Mystery1

Trace: mystery1(3,7)

- c = a - b → c = 3 - 7 = -4
- d = (c >> 7) & 1 →

```
-4 =      11111100
shift 7 = 11111111
        = -1
& 1     = 00000001
d = 1
```

- a - c * d = 3 - (-4) * 1 = 7

Trace: mystery1(8,7)

- c = a - b = 8 - 7 = 1
- d = (c >> 7) & 1 →

```
1 =       00000001
shift 7 = 00000000 = 0
& 1     = 00000000
d = 0
```

- a - c * d = 8 - (1) * 0 = 8

② Mystery2

Trace: mystery2([1,2,3,4], 0, 3)

```
v[0] = v[0] ^ v[3]      v[0] = 0101
v[3] = v[0] ^ v[3]      v[3] = 0001
v[0] = v[0] ^ v[3]      v[0] = 0100
```

values = [4,2,3,1]

Trace: mystery2([1,2,3,4], 1, 2)

```
v[1] = v[1] ^ v[2]      v[1] = 0001
v[2] = v[1] ^ v[2]      v[2] = 0010
v[1] = v[1] ^ v[2]      v[1] = 0011
```

values = [1,3,2,4]

③ Mystery3

Trace: mystery3(5,7)

s = x ^ y ⇒ 0101 ^ 0111 = 0010

c = x & y ⇒ 0101 & 0111 = 0101

1000 ^ 0100 = 1100 = 12

Trace: mystery3(2,8)

s = x ^ y ⇒ 0010 ^ 1000 = 1010

c = x & y ⇒ 0010 & 1000 = 0000

0010 ^ 1000 = 1010 = 10