Name: Breona Pizzuta Date: 9/25/24

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 (give the trace results as numbers in base 10). What does each function do (give a high-level summary)?
Requirement: You should assume integers are only **8 bits** for the purpose of this exercise. The sign bit is the leftmost of the 8 bits.

```
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**
Trace: mystery1(8, 7) returns **8**

Summary: This function returns the maximum of the two parameters.

```
int mystery2(int x) {
    return (x && !(x & (x - 1)));
}
```

Trace: mystery2(1) returns 1
Trace: mystery2(2) returns 1
Trace: mystery2(3) returns 0
Trace: mystery2(4) returns 1
Trace: mystery2(5) returns 0
Trace: mystery2(6) returns 0
Trace: mystery2(7) returns 0
Trace: mystery2(8) returns 1

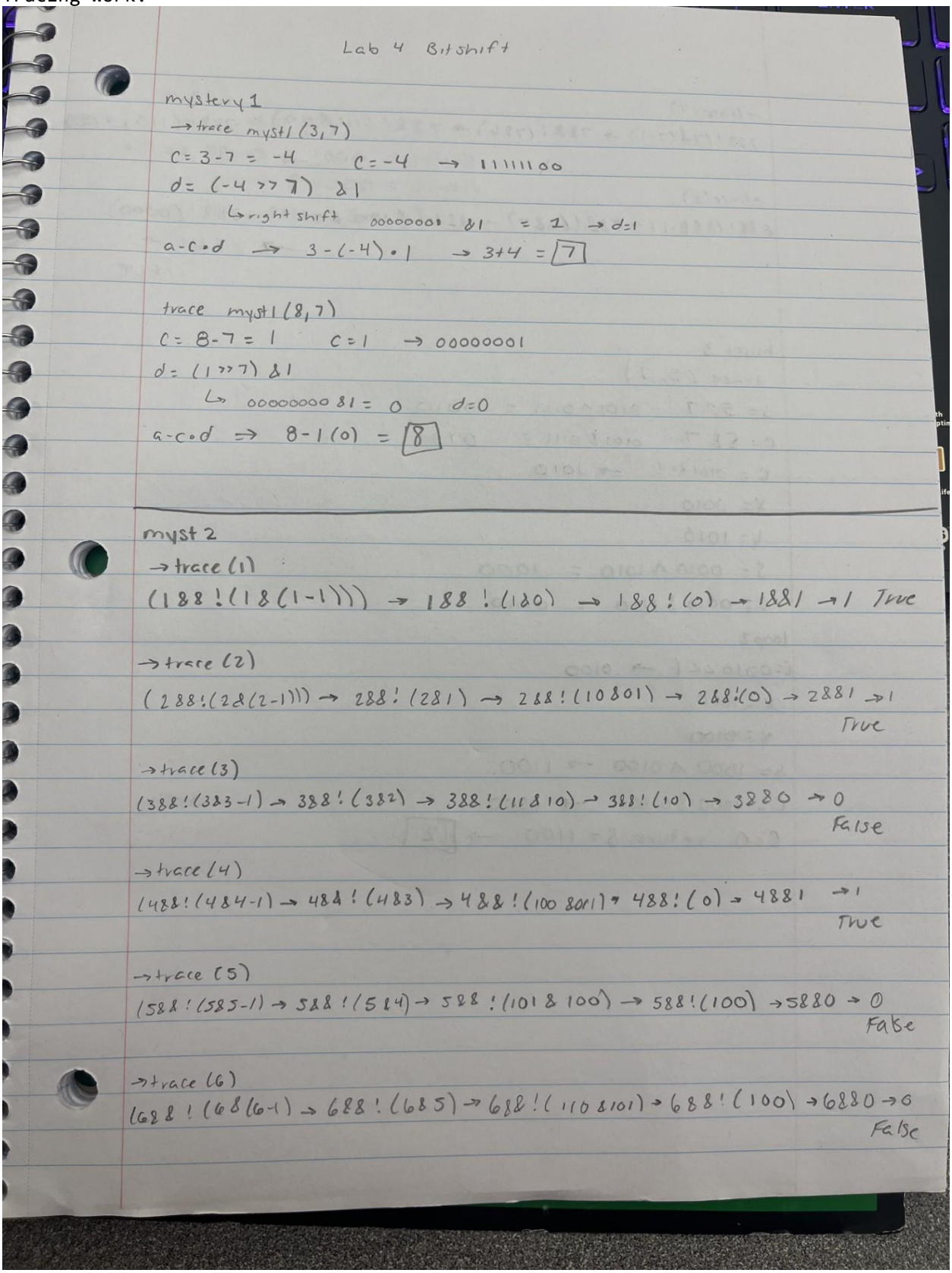
Summary: This function will return 1 (true) if x is a power of 2 and 0 (false) if it is not a power of 2.

```
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**
Trace: mystery3(2, 8) returns **10**

Summary: This function returns the sum of the two parameters.

Tracing work:



Lab 4 Bitshift

mystery1

→ trace myst1(3,7)

c = 3-7 = -4 c=-4 → 11111100

d = (-4 >> 7) &1

↳ right shift 00000001 &1 = 1 → d=1

a-c*d → 3-(-4)*1 → 3+4 = 7

trace myst1(8,7)

c = 8-7 = 1 c=1 → 00000001

d = (1 >> 7) &1

↳ 00000000 &1 = 0 d=0

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

myst 2

→ trace (1)

(188!(18(1-1))) → 188!(180) → 188!(0) → 1881 → 1 True

→ trace (2)

(288!(28(2-1))) → 288!(281) → 288!(10801) → 288!(0) → 2881 → 1

True

→ trace (3)

(388!(383-1) → 388!(382) → 388!(11810) → 388!(10) → 3880 → 0

False

→ trace (4)

(488!(484-1) → 484!(483) → 488!(100 8011 → 488!(0) → 4881 → 1

True

→ trace (5)

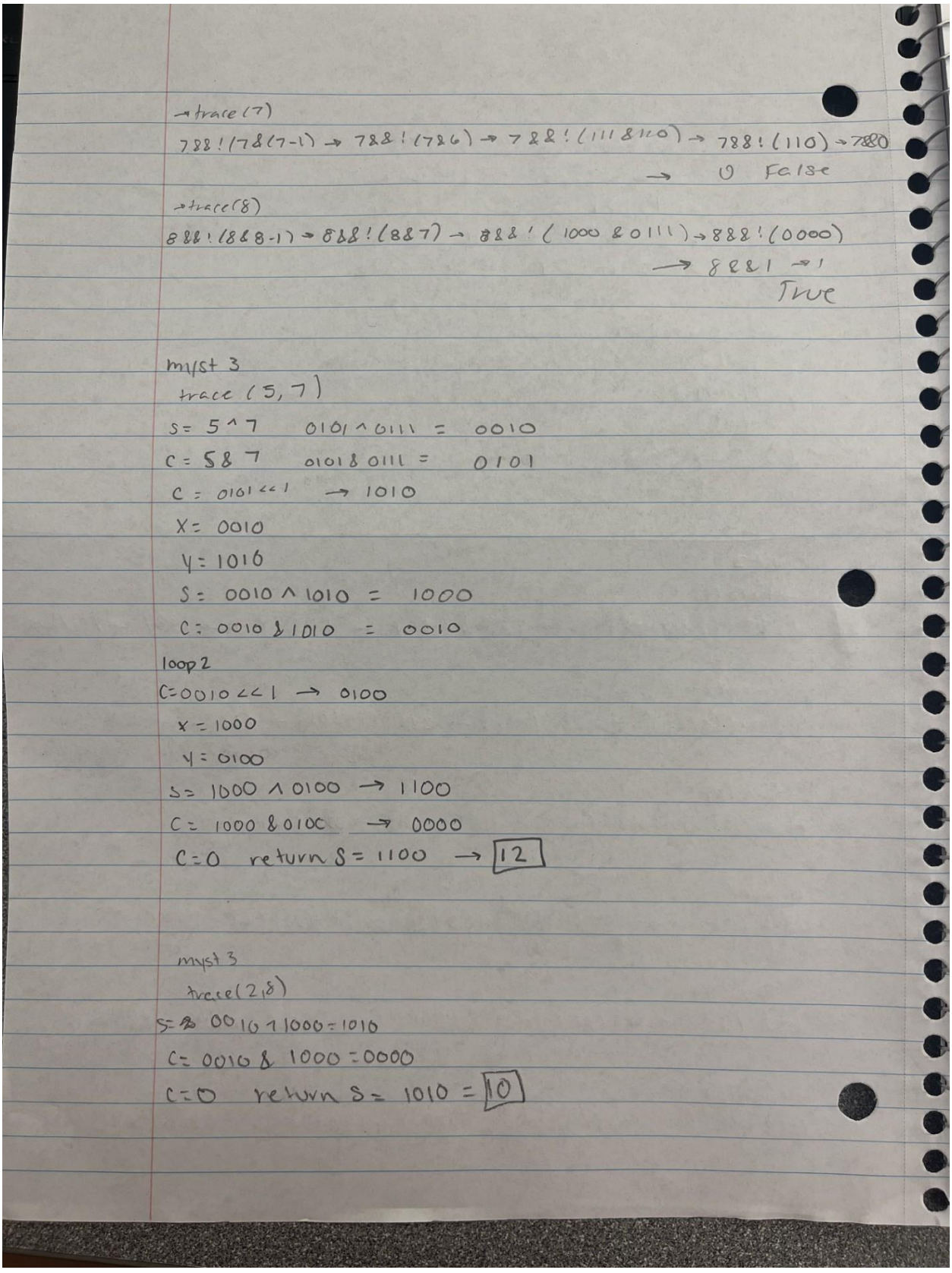
(588!(585-1) → 588!(584) → 588!(101 & 100) → 588!(100) → 5880 → 0

False

→ trace (6)

(688!(68(6-1) → 688!(685) → 688!(110 &101) → 688!(100) → 6880 → 0

False



→ trace(7)

788!(78&(7-1)) → 788!(726) → 788!(111&110) → 788!(110) → 7880

→ 0 False

→ trace(8)

888!(88&8-1) → 888!(887) → 888!(1000 & 0111) → 888!(0000)

→ 8881 → 1

True

myst 3

trace (5, 7)

s = 5 ^ 7 0101 ^ 0111 = 0010

c = 5 & 7 0101 & 0111 = 0101

c = 0101 << 1 → 1010

x = 0010

y = 1010

s = 0010 ^ 1010 = 1000

c = 0010 & 1010 = 0010

loop 2

c = 0010 << 1 → 0100

x = 1000

y = 0100

s = 1000 ^ 0100 → 1100

c = 1000 & 0100 → 0000

c = 0 return s = 1100 → 12

myst 3

trace(2, 8)

s = 0010 ^ 1000 = 1010

c = 0010 & 1000 = 0000

c = 0 return s = 1010 = 10