CSci223
Spring 2005
Homework #2
Due: February 16th

1) Write C code that creates an `int` bit mask that only has the most significant byte set. (3 points)

2) Add the two binary numbers together: (3 points)

```
    0 0 1 0 1 0 1 0
  + 0 1 1 0 0 1 0 1
  -----------------
```

3) Add the two hexadecimal numbers together: (3 points)

```
    0x21C2
  + 0x499F
  --------
```

4) Multiply the two hexadecimal numbers together: (4 points)

```
    0x34A6
  x   0x32
  --------
```

5) Multiply the two binary numbers together: (4 points)

```
    0 0 0 1 1 0 0 1
  x 0 0 0 0 1 0 1 0
  ---------------
```

6) Did overflow occur in the above multiplication? (2 points)

7) How many different values can a 16-bit variable take on? (3 points)

8) What is the largest positive number a 64-bit `int` can represent? (4 points)

9) What is the maximum number of bits needed to represent an 8-digit hexadecimal number? (4 points)

10)What is the hexadecimal representation for `-1` when stored in a 32-bit `int`? (3 points)

11) Why is it wise to avoid unsigned numbers when programming? (4 points)

12) Show how `-567` is represented as a 16-bit `short`. (4 points)

13) The hexadecimal value 0xFFF2 is -14_{10} when interpreted as a two's-complement number. What number does it represent when interpreted as unsigned? (4 points)

14) Show the resulting bit vector when the below 16-bit `short` is assigned to a 32-bit `int`. (5 points)

```
0xF25C
```

15) If two *n*-bit numbers are added together, what is the maximum number of bits that will be needed to contain the result? (3 points)

16) If two *n*-bit numbers are multiplied together, what is the maximum number of bits that will be needed to contain the result? (3 points)

17) How does a number change at the bit-level when cast from a unsigned number to a signed number in C? (5 points)

18) Explain what happens when unsigned addition overflows. (5 points)

19) Write C code to detect when signed addition has overflowed. (7 points)

20) How can we multiply a variable by 6 without using the multiplication operator? (5 points)

21)Define the three fields in an IEEE floating point number. (9 points)

22) Explain denormalized values. (5 points)

23) What are the two special values that can be represented by floating point numbers? Explain when they can be encountered. (4 points)

24) Is it a coincidence that the binary representation for zero is all 0's for unsigned and two's-complement integers as well as floating point numbers? Why might designers have wanted zero to be represented as all zeros? (4 points)