1. **GIVE TWO EXAMPLES** of unary arithmetic operations (NOT operators).
   (a) 
   (b) 

2. For the two examples of unary arithmetic operations, above, **WHAT ARE THE ASSOCIATED OPERATORS?**
   (a) 
   (b) 

3. **GIVE TWO EXAMPLES** of binary arithmetic operations (NOT operators).
   (a) 
   (b) 

4. For the two examples of binary arithmetic operations, above, **WHAT ARE THE ASSOCIATED OPERATORS?**
   (a) 
   (b) 

5. **WHAT IS THE DIFFERENCE** between dividing an `int` by an `int` and dividing a `float` by a `float`? **BE SPECIFIC AND USE THE CORRECT TERMINOLOGY.**

6. **WHAT IS THE DIFFERENCE** between dividing an `int` by a `float` and dividing a `float` by a `float`? **BE SPECIFIC AND USE THE CORRECT TERMINOLOGY.**

7. **WHY** is the use of numeric literal constants in the body (execution section) of a program considered to be bad programming practice?
8. **WHAT IS THE OUTPUT** of each of these programs? Examine the programs CAREFULLY. If a program won’t compile, mark **WON’T COMPILE** and **EXPLAIN**. If a program compiles and runs but does not produce any output, mark **NO OUTPUT** and **EXPLAIN**. If a program compiles and runs but produces garbage output, mark **GARBAGE** and **EXPLAIN**. If you are not confident of an answer, type in, compile and run the program.

(a) #include <stdio.h>

```c
int main ()
{
    int a = 5, b = 7, c;
    a = a + 5;
    c = a * b;
    printf("a = %d, b = %d, c = %d\n", a, b, c);
} /* main */
```

(b) #include <stdio.h>

```c
int main ()
{
    int a = 5, b = 7, c;
    b = b * 5;
    c = a + b;
    printf("%d %d %d\n", a, b, c);
} /* main */
```

(c) #include <stdio.h>

```c
int main ()
{
    float f = 1.0, g;
    int a = 5, b = 4, c, d;
    c = a / b;
    d = b / a;
    g = a / (f * b);
    printf("a = %d, b = %d, c = %d, d = %d\n", a, b, c, d);
    printf("f = %f, g = %f\n", f, g);
} /* main */
```
A C program has the following declarations:

```c
float x = 10.0, y = 5.5, z = 2.1;
int i = 3, j = 5, k = 7, m;
```

**Evaluate** each of the following expressions. **Show Your Work**, including the type of each subexpression (indicating a float with a decimal point). If the expression would compile and runs but would produce garbage output, mark **GARbage** and **Explain**. If you are not confident of your answer, type in, compile and run an appropriate program.

(a) \( y \div x \)

(b) \( i \% j \)

(c) \( k + z \)

(d) \( 100 \div i - x \)

(e) \( 324 \div m - 12 \)

(f) \( i \% (j - 3) \% 3 \)

If you use **Any** resources other than Dr. Neeman, the TAs (Badre, Borah, Sadri, Saravanan), the course textbook or the materials posted on the course webpage, you **MUST** reference them on the quiz. **This includes classmates, friends, professors, online resources, etc.**