1. Evaluate the following expressions:
   a) `36 / 5` → `7`
   b) `18  32 / 6 * 3` → `3`
   c) `6  8 % 11` → `-2`
   d) `22.0 / 5` → `4.4`
   e) `25  7 % 3 + 8 / 3` → `26`
   f) `18.0 + 5.0 * 3.0 / 4.0` → `21.75`

2. Suppose `x`, `y`, and `z` are `int` variables and `x = 2`, `y = 5`, and `z = 6`. What is the output of each of the following statements?
   a) `cout << "x = " << x << ", y = " << y << ", z = " << z;`
      `x = 2, y = 5, z = 6`
   b) `cout << "Sum of " << x << " and " << z << " is " << x + z;`
      `Sum of 2 and 6 is 8`
   c) `cout << "2 times " << x << " = " << 2 * x;`
      `2 times 2 = 4`

3. The following program has syntax mistakes. Correct them. On each successive line, assume that any preceding error has been corrected.

```
const char = STAR = '*'
const int PRIME = 71;
int main
{
    int count, sum;
    double x;
    count = 1;
    sum = count + PRIME;
    x := 25.67;
    newNum = count * ONE + 2;
    sum + count = sum;
    x = x + sum * COUNT;
    cout << " count = " << count << ", sum = " << sum
        << ", PRIME = " << Prime << endl;
}
```

```
#include <iostream>              // include appropriate header
using namespace std;

const char STAR = '*';           // no "=" after char and semicolon at the end
const int PRIME = 71;

int main()                       // () after the main function name
{
    int count, sum;
    double x;
    int newNum;                  //declare newNum

    count = 1;
    sum = count + PRIME;
    x = 25.67;                   // := should be =
    newNum = count * 1 + 2;      // ONE is not declared, should be 1
    sum = sum + count;           // addition cannot be on the left of assignment
    x = x + sum * count;         // COUNT should be count
    cout << " count = " << count << ", sum = " << sum
        << ", PRIME = " << PRIME << endl;     // variable Prime should be PRIME
    return 0;
}
```

4. Suppose a, b, and c are int variables and a = 5 and b = 6. What value is assigned to each variable after each statement executes? If a variable is undefined at a particular statement, report UND (undefined).

	a	b	c
a = (b++) + 3;	9	7	UND
c = 2 * a + (++b);	9	8	26
b = 2 * (++c) - (a++);	10	45	27