Names: ________________________________________________________________

In this lab, we will practice predicting the behavior of programs using our mental models of computation: the substitutive model and the stack model.

**Expression Evaluation**

For each of the following expressions, evaluate them step-by-step to a final value. Write down (a) each step that you take and (b) the final value. Some expressions do not compile; if this is the case, then write this down. Check your work by writing a program that prints the results of the expression using printf.

1. \( 3 + 5 - 2 \)

2. \( 6 - 10 / (2 + 3) \)

3. \( 1 + 5 \% 3 * 2 \)
1 + 5 / 10

1.0 + 5.0 / 10.0

1 + 5.0 / 10.0

1.0 + 5 / 10

1 + 5.0 / 10
Stack Management

For each of the following programs, give the state of the stack and what has been outputted to the console at each of the commented program points. Make sure to reproduce your diagrams exactly for each program state!

(Hint: the labels are in order of where execution flows through the program!)

void f1(void) {
    printf("0");
    // Point (B, D)
}

void f2(void) {
    printf("1");
    // Point (C)
    f1();
    printf("2");
}

int main(void) {
    // Point (A)
    printf("3");
    f1();
    f2();
    // Point (E)
}
```c
void f1(void) {
    printf("0");
    // Points (A, C, E)
}

void f2(void) {
    // Points (B, D)
    printf("1");
    f1();
}

void f3(void) {
    f3();
    printf("2");
}

int main(void) {
    printf("3");
    f1();
    f2();
    f3();
    // Point (F)
}
```