CS 165 Black Box Testing Worksheet

Write a JUnit test class and methods to perform Black Box Testing given the description of the method **sin** in the **Code** class.

```java
public static int sin(int x,
    boolean square,
    boolean increment,
    boolean negate)

Returns the x value with specified modifications. Performs the following operations:
- x is squared if square is true.
- Then x is incremented if increment is true.
- Then x is negated if negate is true.
- x is unmodified otherwise.

Note that the operations are cumulative: negate implies increment and increment implies squared. So if negate is true, the x value will be squared, incremented, and negated in that order, ignoring the values of increment and square.

Parameters:
x - an integer value that is modified based on the other parameters
square - a boolean
increment - a boolean
negate - a boolean

Returns:
the value with the specified modifications.
```
CS 165 White Box Testing Worksheet

Write a JUnit test class to perform Black Box Testing given this implementation of the Code class with method sin.

- Draw the Control Flow Graph using the statement line numbers.
- Create three different test methods, one each for statement coverage, branch coverage, and path coverage.

```java
public static int sin(int x,
                     boolean square,
                     boolean increment,
                     boolean negate) {
    if ( square || increment || negate ) {
        x *= x;
        if ( increment || negate ) {
            x++;
            if ( negate ) {
                x = -x;
            }
        }
    } return x;
}
```