Name:
Date:

// Vocabulary: Nested, Repetition

Now that you know about repetition we can talk about ways to put code inside of other code, which is called nesting, and in fact most loops are nested loops since they are inside of the original loop( ) function. It's easy, all you do is put your loop inside the curly brackets of another loop. Nested if statements work exactly the same way as nested loops.

Example of nested loop:

```
void loop ( ) {
int x = 0;
while (x < 10) {
x  =  x + 1;
}
}
```

Example of nested if statement:

```
if (int x < 10) {
if ( x == 5 ) {
//code here happens if x < 10 & x = 5
}
//code here happens if x < 10
}
```

Imagine your loop( ) racetrack with another for loop racetrack attached to it. This way each time the robot runs (or drives or whatever) around the racetrack it must stop when it reaches a new while loop, run around that race track until that while loop is over and then it can continue running around the larger loop( ) racetrack.

The robot has to run through the whole while loop before it can continue running around the larger loop racetrack. But let's break it down a little more; x starts as zero, if x is less than ten the robot continues running around the while loop until x is not less than ten. If the robot is adding one to x each time it checks the while loop then the robot must run around the while loop a total of ten times. The robot then exits the while loop and continues around the loop racetrack. Next time around the racetrack the variable x will be set to zero again just before the while loop. So, you don't have to worry about the while loop not working due to x being more than or equal to ten.

You can nest as many loops inside of other loops as you like, just make sure you don't get stuck inside of a loop. One way to do this is to misplace curly brackets, so make sure they're in the right spot. If this happens your computer or Arduino will just freeze and you won't really be able to tell why.

Nesting works for code other than just loops! You can nest if statements, loops and many other code structures. All you need to remember is that nesting is a complicated way to say "put code inside of other code" and that the computer eventually needs to get out of the nested statements and back to the loop( ) function so everything can start over again.