Continue and Break

Sometimes a programmer wants to leave the loop body early, rather than finishing all of the statements in side of it. There are two possible behaviors that a programmer might want when leaving the loop body early.

Break

One behavior would be to exit the loop completely, making the execution arrow jump to immediately after the close curly brace which ends the loop (the same place that it goes when the loop's condition evaluates to false). This behavior is obtained by using the break; statement—which we have already seen in the context of switch/case. Whenever the execution arrow encounters a break statement, it executes the statement by jumping out of the innermost enclosing loop (whether it is a while, do-while, or for loop), or switch statement. If the break statement is inside multiple of these which are nested together (e.g. a loop inside a case of a switch statement), then it exits only the most immediately enclosing one. If a break statement occurs and is not inside one of these loops or a switch statement, it is an error in the program.

Continue

The other possible behavior that the programmer might want to have is for the execution arrow to jump back to the top of the loop. This behavior is accomplished with the continue; statement. Executing the continue statement jumps to the top of the innermost enclosing loop (if it is not in a loop, it is an error). In the case of a for loop, the "increment statement" in the for loop is executed immediately before the jump. This fact complicates the de-sugaring of a for loop into a while loop slightly relative to the explanation given above. If the for loop contains any continue statements, then the "increment statement" is written not only before the close curly brace of the loop, but also before any continue statements.

The *Execution of Continue* lecture shows how to transform a for loop with a continue statement inside of it into an equivalent while loop, then execute the resulting code. We note that in this example, simply using an if/else statement would be better—however, a good example of the use of continue is not easy to come by until we learn some more advanced concepts.