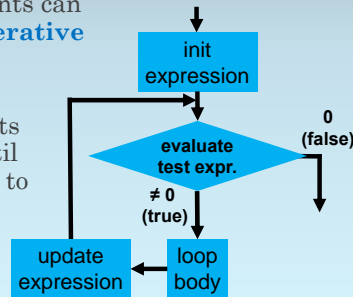


## Simple Statements Can Also Be Iterations

Simple statements can also describe **iterative** execution.

This type of execution repeats a statement until a test evaluates to false (0).



## C's **for** Loop Enables Iterative Execution

The following is called a **for** loop:

```
for (<init>; <test>; <update>) {
    /* loop body */
}
```

As shown on the previous slide, the computer:

1. Evaluates **<init>**.
2. Evaluates **<test>**, and stops if it is false (0).
3. Executes the **loop body**.
4. Evaluates **<update>** and returns to **Step 2**.

## Iterations are Used for Repeated Behavior

```
/* Print multiples of 42 from
   1 to 1000. */
int N;
for (N = 1; 1000 >= N; N = N + 1) {
    if (0 == (N % 42)) {
        printf ("%d\n", N);
    }
}
```

## Let's See How This Loop Works

```
/* Print 20 Fibonacci numbers. */
int A = 1; int B = 1; int C; int D;
for (D = 0; 20 > D; D = D + 1) {
    printf ("%d\n", A);
    C = A + B;
    A = B;
    B = C;
}
```