

Major difference between for and while loop is at pragmatic level because under the hood, both loops are all the same conditional goto; therefore the choice between while and for is arbitrary, based on which seems clearer. Both for and while loops are entry controlled loops that means test condition is checked for truth while entering into the loop's body. The for is usually appropriate for loops in which the initialization and increment are single statements and logically related, since it is more compact than while and it keeps the loop control statements together in one place. The for loop seems most appropriate when number of iteration are known in advance, for example, counting array elements. But, there could be many complex problems where number of iterations depend upon a certain condition and can't be predicated beforehand, in those situation programmers usually prefer to use while loop. The following table lists differences between for and while loop.

Difference Between <code>for</code> and <code>while</code> loops	
for	while
<p>Syntax of for loop: <code>for(initialization; condition; iteration)</code> <code>{</code> <code>// body of the loop</code> <code>}</code></p> <p>The initialization is an assignment statement that is used to set the loop control variable. The condition is a relational expression that determines when the loop exits. The increment defines how the loop control variable changes each time the loop is repeated. The body of loop can either be empty or a single statement or a block of statements.</p> <p>The</p> <code>for(initialization; condition; iteration)</code> <code>{</code> <code>// body of the loop</code> <code>}</code> <p>is equivalent to</p> <code>initialization;</code> <code>while(condition)</code> <code>{</code> <code>// body of the loop iteration;</code> <code>}</code>	<p>Syntax of while loop: <code>while(condition)</code> <code>{</code> <code>// body of the loop</code> <code>}</code></p> <p>The condition may be any expression, and true is any non-zero value. The loop iterates while the condition is true. When the condition becomes false, program control passes to the line of code immediately following the loop.</p>
Pragmatic use of for loop: the for is preferable when there is a simple initialization and increment;	Pragmatic use of while loop: on the other hand while is preferable when the initialization is

since it keeps the loop control statements close together and visible at the top of the loop. This is most obvious in

```
for (i = 0; i < n; i++) {  
    ...  
}
```

which is most often seen while processing array elements.

not just a simple assignment operation and so like increment.

For example, in

```
while ((c = getchar()) == ' ' || c == '\n' || c == '\t');  
/* skip white space characters */
```

there is no initialization or re-initialization, so the while is most natural.