Plausible But Bad Probabilistic Reasoning
Don Knuth, 30 December 2019

In the following program, $U$ stands for a random number, uniformly distributed between 0 and 1, generated independently each time it appears.

\begin{verbatim}
p ← 1
x ← 0
if U < p_1 set x ← 1 and p ← p_1; else set p ← (1 - p_1)p
if U < p_2 set x ← 2 and p ← p_2p; else set p ← (1 - p_2)p
...
if U < p_n set x ← n and p ← p_np; else set p ← (1 - p_n)p
printf("With probability %f I've got x=%d\n", p, x)
\end{verbatim}

Clearly $p$ is the probability that the program has taken the particular sequence of branches that led up to the print statement. But $p$ is not the probability that $x$ has its final value, unless $x \leq 1$! (Because that final value could have been obtained in different ways.)

That flaky reasoning led to a bug in a program that I wrote yesterday.
For example, when $n = 3$ there are eight possible runs. Let $q_j = 1 - p_j$.

<table>
<thead>
<tr>
<th>probability</th>
<th>final $x$ is</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_1p_2p_3$</td>
<td>3</td>
</tr>
<tr>
<td>$p_1p_2q_3$</td>
<td>2</td>
</tr>
<tr>
<td>$p_1q_2p_3$</td>
<td>3</td>
</tr>
<tr>
<td>$p_1q_2q_3$</td>
<td>1</td>
</tr>
<tr>
<td>$q_1p_2p_3$</td>
<td>3</td>
</tr>
<tr>
<td>$q_1p_2q_3$</td>
<td>2</td>
</tr>
<tr>
<td>$q_1q_2p_3$</td>
<td>3</td>
</tr>
<tr>
<td>$q_1q_2q_3$</td>
<td>0</td>
</tr>
</tbody>
</table>

If I really want to exhibit the probability of a particular $x$, I should write this:

\begin{verbatim}
p ← 1
x ← 0
if U < p_1 set x ← 1 and p ← p_1; else set p ← (1 - p_1)p
if U < p_2 set x ← 2 and p ← p_2p; else set p ← (1 - p_2)p
...
if U < p_n set x ← n and p ← p_np; else set p ← (1 - p_n)p
printf("I've set x to %d with probability %f\n", x, p)
\end{verbatim}