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.

$p \leftarrow 1$
$x \leftarrow 0$
if $U < p_1$ set $x \leftarrow 1$ and $p \leftarrow p_1 p$; else set $p \leftarrow (1 - p_1)p$
if $U < p_2$ set $x \leftarrow 2$ and $p \leftarrow p_2 p$; else set $p \leftarrow (1 - p_2)p$
$\vdots$
if $U < p_n$ set $x \leftarrow n$ and $p \leftarrow p_n p$; else set $p \leftarrow (1 - p_n)p$
printf(`"With probability %f I've got x=%d\n"`, p, x)

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 $q_j = 1 - p_j$.

probability	final x is
$p_1 p_2 p_3$	3
$p_1 p_2 q_3$	2
$p_1 q_2 p_3$	3
$p_1 q_2 q_3$	1
$q_1 p_2 p_3$	3
$q_1 p_2 q_3$	2
$q_1 q_2 p_3$	3
$q_1 q_2 q_3$	0

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

$p \leftarrow 1$
$x \leftarrow 0$
if $U < p_1$ set $x \leftarrow 1$ and $p \leftarrow p_1$; else set $p \leftarrow (1 - p_1)p$
if $U < p_2$ set $x \leftarrow 2$ and $p \leftarrow p_2$; else set $p \leftarrow (1 - p_2)p$
$\vdots$
if $U < p_n$ set $x \leftarrow n$ and $p \leftarrow p_n$; else set $p \leftarrow (1 - p_n)p$
printf(`"I've set x to %d with probability %f\n"`, x, p)