REMARK ON ALGORITHM 299

Chi-Squared Integral [I. D. Hill and M. C. Pike, *Commun. ACM 10*, 4 (Apr. 1967), 243–244.]

I. D. Hill and M. C. Pike [Received February 1985; accepted February 1985] Clinical Research Centre, Harrow, Middlesex HA1 3UJ, England and ICRF Cancer Epidemiology and Clinical Trials Unit, Radcliffe Infirmary, Oxford OX2 6HE England

This algorithm included a **Boolean** parameter *bigx* to be set to **true** if the value of x is too big for $exp(-0.5 \times x)$ to be accurately represented by the machine, or **false** otherwise. The aim was to avoid overflow and underflow problems, but these can still arise in some cases.

The following changes are therefore recommended:

(1) Change **real** x; to **real** x, *bigx*;
(2) Delete **Boolean** *bigx*;
(3) After **Boolean** *even*; insert
```
real procedure ex(x);
value x; real x;
ex := if x < - bigx then 0.0 else exp(x);
```
(4) Change
```
if even ∨ f > 2 ∧ ¬ bigx then y := exp(-a)
```
to
```
if f > 1 then y := ex(-a)
```
(5) Change **if** *bigx* **then** to **if** $a >$ *bigx* **then**
(6) Change $s := exp(c \times z - a - e) + s$
to
$$s := ex(c \times z - a - e) + s$$
(7) Change $e := e \times a/z$; to $e := e \times (a/z)$;
(8) Change the introductory comment to say,
"The parameter *bigx* should be set to such a value that *exp* (−*bigx*) will just not underflow. This will normally be a constant for any given model of computer and could be set as an internal constant instead of being passed each time as a parameter, if desired."