# P4 CR for rootn case differs from IEEE 754

**WG 14 N2309**
**2018-10-09**
**C FP Group**

**TS 18661-4 CR nn**
==========

**Reference Document: TS 18661-4**

**Subject: rootn case differs from IEEE 754**

**Summary**

IEEE 754-2008 (IEC 60559:2011) neglected to specify infinity cases for its rootn operation. TS 18661-4 added these cases. Later the IEEE 754-2019 revision also added them but with one case different: rootn(-inf, n) for even n > 0. TS 18661-4 says the result is the same as rootn(-0, -n) without a divide-by-zero floating-point exception, which TS 18661 and IEEE 754 agree is +inf. IEEE 754-2019 says the result is qNaN with an invalid exception.

The following suggested TC changes rootn in TS 18661-4 to match IEEE 754-2019.

**Suggested Technical Corrigendum**

In TS 18661-4, clause 7, in C F.10.4.8, change:

— **rootn(**±∞, *n***)** is equivalent to **rootn(**±0, −*n***)** for *n* not 0, except that the "divide-by-zero" floating-point exception is not raised.

to:

— **rootn(**+∞, *n***)** is +∞ for *n* > 0.
— **rootn(**−∞, *n***)** is −∞ for odd *n* > 0.
— **rootn(**−∞, *n***)** is qNaN and raises the "invalid" floating-point exception for even *n* > 0.
— **rootn(**+∞, *n***)** is +0 for *n* < 0.
— **rootn(**−∞, *n***)** is −0 for odd *n* < 0.
— **rootn(**−∞, *n***)** is qNaN and raises the "invalid" floating-point exception for even *n* < 0.