

FLP01-C. Take care in rearranging floating-point expressions

Be careful when rearranging floating-point expressions to ensure the greatest accuracy of the result.

Subclause 5.1.2.3, paragraph 14, of the C Standard [ISO/IEC 9899:2011], states:

*Rearrangement for floating-point expressions is often restricted because of limitations in precision as well as range. The **implementation** cannot generally apply the mathematical associative rules for addition or multiplication, nor the distributive rule, because of roundoff error, even in the absence of overflow and underflow. Likewise, implementations cannot generally replace decimal constants to rearrange expressions. In the following fragment, rearrangements suggested by mathematical rules for real numbers are often not valid.*

```
double x, y, z;  
/* ... */  
x = (x * y) * z; /* not equivalent to x *= y * z; */  
z = (x - y) + y; /* not equivalent to z = x; */  
z = x + x * y; /* not equivalent to z = x * (1.0 + y); */  
y = x / 5.0; /* not equivalent to y = x * 0.2; */
```

Risk Assessment

Failure to understand the limitations in precision of floating-point-represented numbers and their implications on the arrangement of expressions can cause unexpected arithmetic results.

Recommendation	Severity	Likelihood	Remediation Cost	Priority	Level
FLP01-C	Low	Probable	High	P2	L3

Related Vulnerabilities

Search for [vulnerabilities](#) resulting from the violation of this recommendation on the [CERT website](#).

Related Guidelines

SEI CERT C++ Coding Standard	FLP01-CPP. Take care in rearranging floating-point expressions
ISO/IEC TR 24772:2013	Floating-point Arithmetic [PLF]

Bibliography

[ISO/IEC 9899:2011]	Subclause 5.1.2.3, "Program Execution"
-------------------------------------	--

