On the limits of time travel in the face of undefined behavior in C

November 4, 2024

Some time ago, I noted that <u>undefined behavior can result in time travel</u>. Specifically, that once undefined behavior occurs, the undefined behavior extends to the entire program, *even the parts that executed before the undefined behavior occurred*.

The citation I made was from the C++ standard, which grants blanket time travel permission. <u>The C standard</u> does impose some limits on how far back in time undefined behavior can extend. Specifically, undefined behavior cannot alter *observable behavior* that has already occurred. The definition of observable behavior appears in section 5.1.2.4.

> The least requirements on a conforming implementation are:
>
> - Volatile accesses to objects are evaluated strictly according to the rules of the abstract machine.
> - At program termination, all data written into files shall be identical to the result that execution of the program according to the abstract semantics would have produced.
> - The input and output dynamics of interactive devices shall take place as specified in 7.23.3. The intent of these requirements is that unbuffered or line-buffered output appear as soon as possible, to ensure that prompting messages appear prior to a program waiting for input.

We can ignore the "At program termination" clause because it says that the behavior must match the abstract semantics, but to reach program termination, the program has already proceeded past undefined behavior, and undefined behavior specifies no abstract semantics, so anything is possible, and that clause imposes no constraints.

That leaves the other two.

Volatile accesses prior to the undefined behavior remain valid, so you cannot time travel-away a volatile access.

And interactive device activity is also not subject to time travel. So anything you printed to the console or read from the console cannot be recalled. This is already true in a metaphysical sense: The output was already printed to the screen. You can't go back and un-print it. (Mind you, the undefined behavior is permitted to *erase* the previous output from the screen.)

But everything else is still in play. You can time-travel away normal (non-volatile) memory accesses and non-interactive file access.