Race condition

A race condition or race hazard is the condition of an electronics, software, or other system where the system's substantive behavior is dependent on the sequence or timing of other uncontrollable events. It becomes a bug when one or more of the possible behaviors is undesirable.

In software

A race condition arises in software when a computer program, to operate properly, depends on the sequence or timing of the program's processes or threads. Critical race conditions cause invalid execution and software bugs Critical race conditions often happen when the processes or threads depend on some shared state. Operations upon shared states are done in critical sections that must be mutually exclusive. Failure to obey this rule can corrupt the shared state.

A race condition can be difficult to reproduce and debug because the end result is nondeterministic and depends on the relative timing between interfering threads. Problems of this nature can therefore disappear when running in debug mode, adding extra logging, or attaching a debugger. It is therefore better to avoid race conditions by careful software design.