EX: Integrated circuits have been known to have faulty mathematical processors. In one case, a company released a chip with a known error that they were hopeful would remain undiscovered for about one year (until the next-generation chip came out). The error occurred in a coprocessor that multiplied numbers.

To illustrate the type of calculation one might do to estimate how often such an error would occur, consider the following fictitious scenario: an error occurs in the 12th digit of the answer when, in two 9-bit binary numbers being multiplied, the 3-bit pattern 101 in bits 1-3, or 4-6, or 7-9 occurs exactly twice. In other words, separate each 9-bit binary number into 3 sets of 3 bits. Then look at each of the six 3-bit patterns. If the pattern 101 occurs exactly twice in the six patterns, the error occurs. Assume all bits are equally likely to be 0 or 1 independent of all other bit values.

Also, we assume that the probability of anyone noticing an error in the 12th digit of a product will be $1/10^{12}$, and the number of multiplies computed each second is 10^8.

How long will it take for P(error noticed) to exceed 1/2?

SOL'N: We solve this problem in several steps. Starting from the outside and working our way in, the result we are really going to solve for is how many multiplies, (call this n), it will take to reach the point where P(error noticed in n trials) = 1/2. The time this takes in seconds will be $n/10^8$.

If we try to calculate P(error noticed in n trials) directly, the calculation gets very messy. It is difficult to count all the ways of noticing an error while avoiding over-counting. Suppose we had $n = 2$, for example. Then the formula for the probability of a union of events would give the probability for noticing an error:

$$\begin{aligned} P(\text{error noticed in 2 trials}) = \; & P(\text{error noticed in 1st trial}) \\ & + P(\text{error noticed in 2nd trial}) \\ & - P(\text{error noticed in both trials}) \end{aligned}$$

Extending the above formula to a large value of n becomes intractable. Instead, we use the complement of the event the error is noticed in n trials:

$$P(\text{error noticed in } n \text{ trials}) = 1 - P(\text{error not noticed in } n \text{ trials})$$

Our task now becomes calculating the following probability:

P(error not noticed in n trials) = 1 – 1/2 = 1/2

The only way to not notice the error in n trials is to not notice it on every one of the n trials. In other words, the error must not be noticed on the 1st trial And not be noticed on the 2nd trial And ... And not be noticed on the nth trial. Suppose we had $n = 2$, for example. Then the formula for the probability of the intersection of independent events would give the probability for noticing an error:

P(error not noticed in 2 trials) = P(error not noticed in 1st trial)
· P(error not noticed in 2nd trial)

NOTE: If the events were not independent, the formula would change to

P(error not noticed in 2 trials) = P(error not noticed in 1st trial)
· P(error not noticed in 2nd trial | error not noticed in 1st trial)

For n large, this would also become intractable, although it is still arguably simpler than the formula for the probability of noticing an error.

For all trials, P(error not noticed in one trial) is the same. Thus, for n trials, we have

P(error not noticed in n trials) = P(error not noticed in one trial)^n.

Now we turn to the problem of finding P(error not noticed in one trial). Here, finding the probability of the complement of this event will be easier. Thus, we will find P(error noticed in one trial), which we may express as

P(error noticed in one trial) = P(error occurs in one trial)P(err noticed)

This formula follows from the independence of an error occurring and an error being noticed when it occurs. The problem statement says P(err noticed) = $1/10^{12}$, and we are left with the problem of computing P(error occurs in one trial). To find this value, we consider the probability of one particular pattern of bits for the two nine-bit words that causes an error. It turns out that any other pattern that causes an error has the same probability. Thus, we can multiply the probability for one error-causing pattern by the number of such error-causing patterns, (found by using a combinatorial coefficient), to find P(error occurs in one trial):

$$P(\text{error occurs in one trial}) = P(\text{particular error pattern}) \cdot \#\text{ error patterns}$$

Suppose the particular error-causing pattern we pick is the one with the first two 3-bit patterns = 101 and the remaining four 3-bit patterns ≠ 101. The probability of the pattern 101 is 1/2 · 1/2 · 1/2 = 1/8 since zeros and ones are equally likely and bit values are independent. It follows that the probability of a pattern that is ≠101 is 1 – 1/8 = 7/8. Since all bits are independent, we have:

$$P(\text{pattern } 101\ 101\ {\neq}101\ {\neq}101\ {\neq}101\ {\neq}101) = \left(\frac{1}{8}\right)^2 \cdot \left(\frac{7}{8}\right)^4$$

The number of error patterns is the number of ways of picking two locations for the 101 patterns out of the six 3-bit segments. This is given by a combinatorial coefficient:

$$\#\text{ error patterns} = {}_6C_2 = \frac{6!}{4!2!} = \frac{6 \cdot 5}{2 \cdot 1} = 15$$

NOTE: We might multiply ${}_6C_2$ by ${}_4C_4$ for the number of ways of picking the locations of the ≠101 patterns after the two 101 patterns are picked, but ${}_4C_4 = 1$. That is, there is only one way to pick where the remaining four ≠101 patterns can go.

We are now ready to calculate P(error occurs in one trial):

$$P(\text{error occurs in one trial}) = \left(\frac{1}{8}\right)^2 \cdot \left(\frac{7}{8}\right)^4 \cdot 15 \approx 0.1374$$

Substituting this into our equation for the probability of noticing an error in one trial, we have

$$P(\text{error noticed in one trial}) = 0.1374 \cdot 10^{-12}.$$

Returning to an earlier formula, we have

$$P(\text{error not noticed in one trial})^n = (1 - 0.1374 \cdot 10^{-12})^n.$$

To find n, we solve the following equation:

$$(1 - 0.1374 \cdot 10^{-12})^n = 1/2$$

The left side of this equation is of the form $(1 - x)^n$ where $x << 1$ is very small. We may, therefore, use an approximation derived from the binomial expansion:

$$(1 - x)^n \approx 1 - nx$$

Using this approximation, the equation we solve is

$$1 - n(0.1374 \cdot 10^{-12}) = 1/2$$

or

$$n(0.1374 \cdot 10^{-12}) = 1/2.$$

This yield the value $n = 3.639 \bullet 10^{12}$. Dividing this by the 10^8 calculations per second gives a time of

$$t = 3.639 \cdot 10^{12} \text{ s} \approx 10.4 \text{ hr} = 10{:}24$$

The error is likely to be noticed in about 10 hours. (In the case of the actual chip, the error was found and publicized within a few days, causing the manufacturer to presumably reconsider its decision.)