Online Ethics Center
FOR ENGINEERING AND SCIENCE

Software Testing

Author(s)

Joseph H. Wujek

Year

1995

Description

A project team is developing new software for airplane altitude controls combined with navigation. How is the software to be tested? How are the results to be interpreted, and what are the expectations and goals for the quality of the software? Suitable for courses in statistics, software engineering, reliability engineering, levels 3-4.

Body

Introduction

You are an engineer employed by Wondrous Avionics, Inc. (WA, Inc.). You are working on the project team developing the Mark-5, a new device in the prototype stage. The Mark-5 is a system for airplane altitude control combined with navigation.

It has 116 input variables: X_1, X_2, ... X_6. Each X_i can take on any of the values permitted by a 32-bit word. Each X_i is sampled simultaneously for 20 nanoseconds every 1.30 milliseconds. Sampling at 1.30 milliseconds is the fastest possible due to the actuation and settling time of the electro-hydraulic mechanisms controlled by the Mark-5. Each of the possible totality of states of the X_i corresponds to one, and only one, configuration of aircraft control surfaces and resultant aircraft altitude, Y_j . Thus, $Y_j = f[X_1, X_2, ... X_{116}]$ in one-to-one correspondence. The Mark-5 outputs one value of Y_j in each 1.30 ms intervals. The output variable Y_j is generated by software, using a program which resides in firmware in the Mark-5. The software result actuates appropriate hardware drivers to actuate the hydraulic mechanisms. In response to concerns from potential users of the Mark-5 regarding the use of software in safety-critical systems, the CEO tells the project team, "The Mark-5 must be tested in all possible states to be sure that the software always works. Our customers are nervous about using software this way. I want us to answer their concerns by demonstrating, by test, that the Mark-5 gives the right output for each combination of inputs. After all, under some conditions the wrong output could cause a plane to crash!"

Back to Top

Questions

1. Assuming that the test speed is limited only by the 1.30 millisecond cycle time of the Mark-5, how long in hours would it take to perform the test desired by the CEO? Assume the test proceeds as fast as possible and without interruption, 24 hours per day, 7 days per week.
2. Same as (a), but assume three bugs were found, and each took two days to find and fix. Assume that the bugs were found at the 1/3, 2/3 and 99% complete points. The test must be run in its entirety after each bug fix.
3. WA estimates that it will cost $700 per hour to run the test. Compute the cost of the test in part (a) and part(b).
4. Suppose it is decided that, instead of a 32-bit word, an 8-bit word is sufficient. Assume because only the software is being tested, not and the integrated system, the test cycle time can be reduced from 1.30 millisecond to 130 nanosecond. Rework part (a) with these design/test changes. (Is this a good assumption to make?)

5. In your view, what would be a "reasonable" test to run on the Mark-5 *system?*
6. Should the Mark-5 be built and offered for sale? What ethical principles support your conclusion?

Back to Top

Solutions

1. This problem is intended to demonstrate the difficulties of a deterministic approach to software testing. A "brute force" analysis of all permutations of states yields an absurdly long test time. Therefore, a probabilistic approach must be employed, coupled with careful estimates of state-occupancies determined by detailed analyses of the input/output states and the software coding. (Doing so is beyond the scope here, but is extremely important in software engineering.) Even with these modern methods, test times are exceedingly long; and the process is expensive and complicated. In the end, one must accept a bounded risk, itself a risk! A 32 bit word can take on 2^{32} = 4.295E9 states (rounding to four SF). Each of the 116 variables may take on any of these values. So, the number of distinct states is: S = $(4.295E9)^{1l6}$. Logarithms may be used to find: S = 2.653(E1117) possible states. The interval between sampling is 1.30 ms, so the test-time is: T = (2.653)(E1117)(1.30E-3) = 3.449(E1114) seconds, or 9.6(E1110) hours. Based on 24 hours/day, 365 days/year testing it would thus require a minimum (no restart from zero after bug fixes) of l.l(E1107) years to perform the 100% test! (The age of the universe is estimated to be of the order of E10 years.)
2. The 2 days/bug *to fix bugs* is negligible compared to the test time, to say the least! Thus the accumulated test time is: T = [9.6(El l lO)hours]t(l/3) + `2J3) + (0.99) + (1)] = 2.9(E1111) hours.
3. 2.9(E1111 h)($7001h) = $2.0E1114.
4. An 8-bit word has 2^8 = 256 states. Then $(256)^{116}$ = 2.27E279 possible states exist. Test time is T = (1.30E-7)(2.27E279) = 2.95E272 seconds, or 8.2E268 hours, still an impossible test!
5. From what is given in the problem statement, no "reasonable" test exists. If it is technically possible to test each word in parallel, then somehow combine results in some manageable form, a test may be possible. But such artifices are dependent upon engineering judgment and may not yield a thorough and

reliable test.

6. Based upon the results above, particularly part (e), the Mark-5 should not be sold. An interesting class discussion could be had on the ethics of building or not-building the *Strategic Defense Initiative* system ("Star Wars"), a system of far more complexity and consequences than the hypothetical Mark-5.

Back to Top

Notes

Author: Dr. Joseph H. Wujek, P.E.

These problems were originally developed as part of an NSF-funded project to create numerical problems that raise ethical issues for use in engineering and other course assignments. The problems presented here have been edited slightly for clarity.

Rights

Resource Type

Case Study / Scenario

Parent Collection

Numerical & Design Problems With Ethical Content

Topics

Data Management
Product Liability
Safety

Discipline(s)

Computer Sciences
Authoring Institution
Zachry Department of Civil Engineering-TAMU Ethics