

Bugs in Software

Object Oriented Software Engineering Lecture 4

Dr. Graham McDonald
graham.mcdonald@glasgow.ac.uk

Outline

- Software reliability
- What is a Bug?
- Why Bugs occur?
- Cost of Bugs
- Bug Repository
- Bug Triaging

What is Software Reliability

- The probability of failure-free software operation for a specified period of time in a specified environment
- It denotes a product's trustworthiness or dependability.

Software Reliability

- Software reliability not caused due to aging but **due to bugs**
- The more the bugs, the lesser the reliability of the software
- Still failures seem random, hence reliability theory can be applied

Examples of Bugs

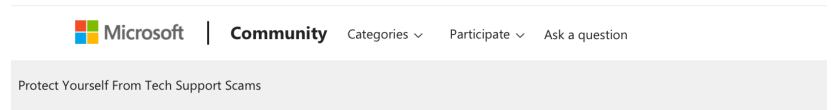
Intel Pentium Floating-Point Division Bug

- Enter the following equation into your PC's calculator:
 $(4195835 / 3145727) * 3145727 - 4195835$
- If the answer is zero, your computer is just fine.
- If you get anything else, you have an Intel Pentium CPU with a floating-point division bug

5

Examples of Bugs

The context of a bug

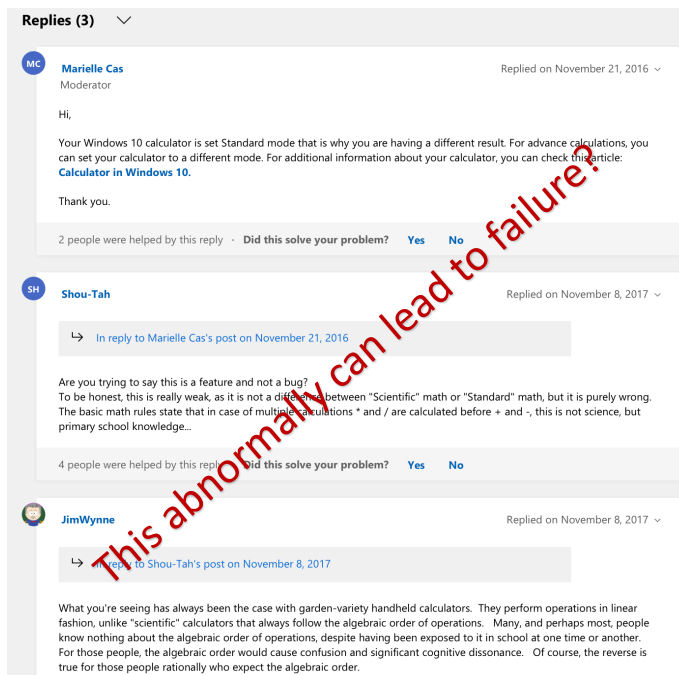


Created on November 20, 2016

windows 10 calculator problem

when i press on windows 10 calculator $1+3*3=12$
how it possible ??
but google calculator $1+3*3=10$ that are normal result
i want to say that there are some bug in windows 10 calculator

6



7

What Is a Bug?

A software bug occurs when one or more of the following five rules is true:

1. The software doesn't do something that the product specification says it should do.
2. The software does something that the product specification says it shouldn't do.
3. The software does something that the product specification doesn't mention.
4. The software doesn't do something that the product specification doesn't mention but should.
5. The software is difficult to understand, hard to use, slow, or—in the software tester's eyes—will be viewed by the end user as just plain not right.

8

Bug

Definition: (Software Bug)

A common term used to describe a flaw, mistake, or failure in a computer system that produces an incorrect or unexpected result, or causes it to behave in unintended ways.

Bugs in Software

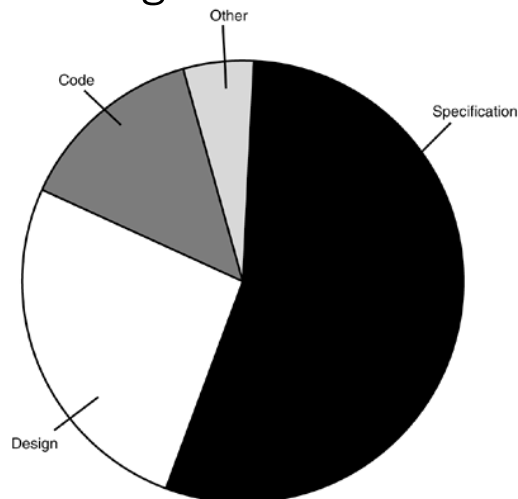
Object Oriented Software Engineering

Lecture 4: Part 2

Dr. Graham McDonald

graham.mcdonald@glasgow.ac.uk

Why Do Bugs Occur?



Why Do Bugs Occur?

1. The number one cause of software bugs is the specification, why?
 - It is not written
 - It is not complete
 - It is not clear
 - It is constantly changing
 - It is not communicated well to the development team

Why Do Bugs Occur?

2. The next largest source of bugs is the design.

- Inappropriate modelling
- Lack of modelling tools
- Time to market pressure

A lot of times developers do not consider design as an important step. Hence unable to identify design bugs introduces as inappropriate consideration of design choices and the alternatives.

13

Code Level Bugs Examples

- Apache Ant 1.6.2,
org.apache.tools.ant.taskdefs.optional.metamata.MAudit

```
if (out == null) {  
    try {  
        out.close();  
    } catch (IOException e) {  
    }  
}
```

Why Do Bugs Occur?

3. Coding errors

- Software complexity
- Poor documentation
- Limited time
- Programmer skills

14

Code Level Bugs Examples

- Eclipse 3.0.1, org.eclipse.update.internal.core.ConfiguredSite

```
if (in == null)  
    try {  
        in.close();  
    } catch (IOException e1) {  
    }  
}
```

Code Level Bugs Examples

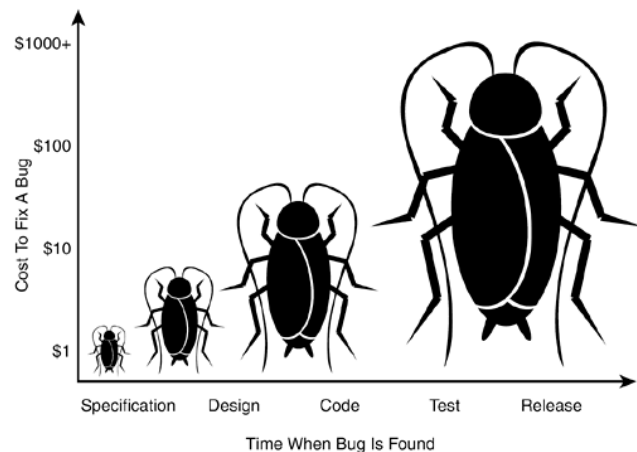
- J2SE version 1.5 build 63 (released version),
java.lang.annotation.AnnotationTypeMismatchException

```
public String foundType() {  
    return this.foundType();  
}
```

Typical Coding Bugs (Defects)

- Improper use of class library
- Inheritance implementation error
- Wrong message
- Wrong object
- Override error
- Data flow error
- Misuse local variable
- Scope boundary error
- Domain knowledge
- Algorithm error
- Need refactoring
- Missing operation
- Unnecessary code
- Syntax error

Cost of Bugs



Cost of Bugs

Most forms of testing only find about **1 bug** out of every **3**. And all tests together barely remove **85%** of bugs during testing

Even the best companies and organizations have released products with expensive (but sometimes simple to fix) bugs

Bug Repository

- Software users and developers report bugs, to allow software developers to fix them.
- Bugs are reported using **bug reports** which are added to an **issue tracking system** or **bug repository**.
 - Apache Bloodhound
 - Bugzilla
 - Jira
 - Trac
 - etc.

21

Bug Triage

- Prioritizing bugs based on their seriousness and deciding what to do about each one
- Fix what really need to be fixed
 - By Design
 - Duplicate
 - Postponed
 - Not Reproducible
 - Won't fixed
 - Reassigned
 - Fixed

Bug Triaging

Definition: (Bug Triaging)

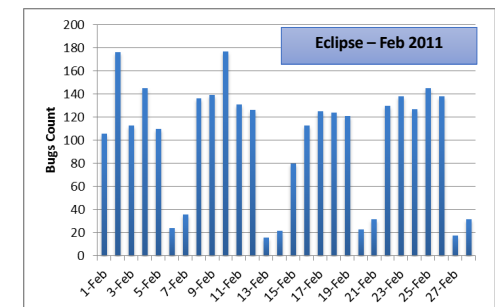
Assigning a bug to the most appropriate/capable developer who will fix it.

- The **bug triager** reads, analyzes, (understands) and assigns bug fixers for each newly reported bug.

Bug Triaging

- Bug triager challenges:
 - Knowledge about the system/project;
 - Descriptiveness of bug report;
 - Rate of reporting bugs;
 - Many developers, different projects, and various expertise!

System	#Fixers
Eclipse	2,144
Firefox	3,014
Jazz	156
Gcc	293
Apache	1,695
FreeDesktop	374
NetBeans	380



Bug Tracking Work Flow

- A tester find a bug and report it
- The bug is assigned to a manager for initial triage
- The manager resolve it or assign to a developer
- Resolved bug returned to a tester
- The tester either closes the bug or reopen it with additional information or comments which start step 2 again

Goal of a software tester

- ... to *find* bugs
- ... as *early* in the software development processes as possible
- ... and make sure they get *fixed*.
- **Advice:** Be careful not to get caught in the dangerous spiral of unattainable perfection.

Conclusion

—Robert Sabourin, *I am a Bug*

You know you are finished when the only bugs left are the ones that you decide you can live with at least for now!

