Software Testing

Introduction to Software Testing

by J. Janvier (Jay/文宇)

What is Software Testing?

The process of verifying and validating
software against the customer’s requirements

Non-conformities found during this process are called:
defects, faults, bugs, flaws, issues, mistake etc.

Process is: an organized or structured way of doing things.

What we will learn in this course:

Structured Software Testing

Software Tester vs. Programmer

Programmer

Understands the system

but, will test "gently"

and, is driven by "*delivery*“ of the software

Software Tester

Must learn about the system,

but, will attempt to **break** it

and, is driven by *quality*

Why Software Testing?

Europe, 1996: Ariane 5 Rocket explodes after launch. Cause: defects in re-used software components Result: 500 million US$ wasted

http://itsfoss.com/a-floating-point-error-that-caused-a-damage-worth-half-a-billion/

Lesson Learned:

We are not able to create 100% defect free software

We are not able to create 100% defect free software **with** structured software testing

Small bugs can have huge consequences

The earlier we find bug, the cheaper it is to fix the mistake

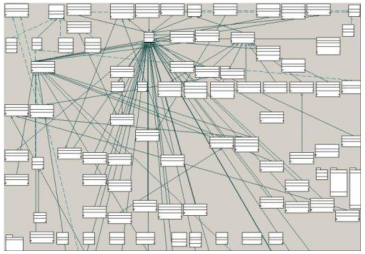
We –normally- test until we reach a planned level of quality (or when money runs out)

Defects can originate due to:

- Human error (throughout the software development lifecycle)
  Example: Mistakes in design specifications, bad code

- Time/Budget pressure
  Example: Reach the market before competion

- Complexity of the product
  Example: Complex architecture, changing interfaces



- New and/or changing Technologies
  Example: Use of new (versions of) libraries, new IDE

Defects can originate due to:

- External dependancies

  Example: Client/Server application depends on internet

- Environmental conditions (Radiation, Magnetic fields, EMI)

  Example: Mobile phones in hospital

- Wrong use/application of product or product parts

- ….

Basic Steps of Structured Testing

1. **Write Test Management Plan**
   - Analysis: Project Plan
   - Estimate: needed time, money, resources, test (prioritization) strategy
2. **Select what has to be tested**
   - Analysis: What is to be delivered (requirements)
   - Design: tested for cohesion (designs)
3. **Decide when, how and to what extent the testing needs to be done**
   - How: What techniques?
   - When: Which phase(s) of the project
   - Extent: Decide on whether or not to automate the tests (regression testing)

Basic Steps of Structured Testing

4. <u>Develop</u> test scenarios/cases

- Develop Test Scenarios
- Develop the Tests

5. <u>Execute</u> the tests

- Select Test Data
- Execute the Tests
- Report the found Defects and follow-up/re-test the Defects

6. Write Test Report(s)

- Analyze the results
- Document the results in a report

Questions?