

# WK3

11/16/2005 4:30 PM

---

## THE VENERABLE TRIANGLE REDUX

---

William Rollison  
Microsoft Corporation

## BJ Rollison

BJ Rollison is a Test Architect with Microsoft's Engineering Excellence Group and has more than 16 years of experience in the software industry. In 1999, he became the Director of Test Training of Microsoft's Internal Technical Training group and was responsible for planning and organizing training for more than 6000 test engineers at Microsoft. Two years ago BJ became a technical trainer in the Engineering Excellence Group where he currently designs, develops, and delivers intensive hands on technical training curriculum for new and experienced test engineers at Microsoft. BJ also teaches software testing courses at the University of Washington and sits on the advisory boards for software testing certificate programs at the University of Washington and Lake Washington Technical College.



# The Venerable Triangle Redux: A familiar problem; a new outlook



---

Bj Rollison, Test Architect  
Engineering Excellence Group  
Microsoft, Inc.  
(willro@microsoft.com)

# My career in software testing



Welcome to testing!



- Domain knowledge
- Experience
- Intuition
- Exploration
- Trial and error
- Unstructured OJT

# But, how much do we really know about software testing?

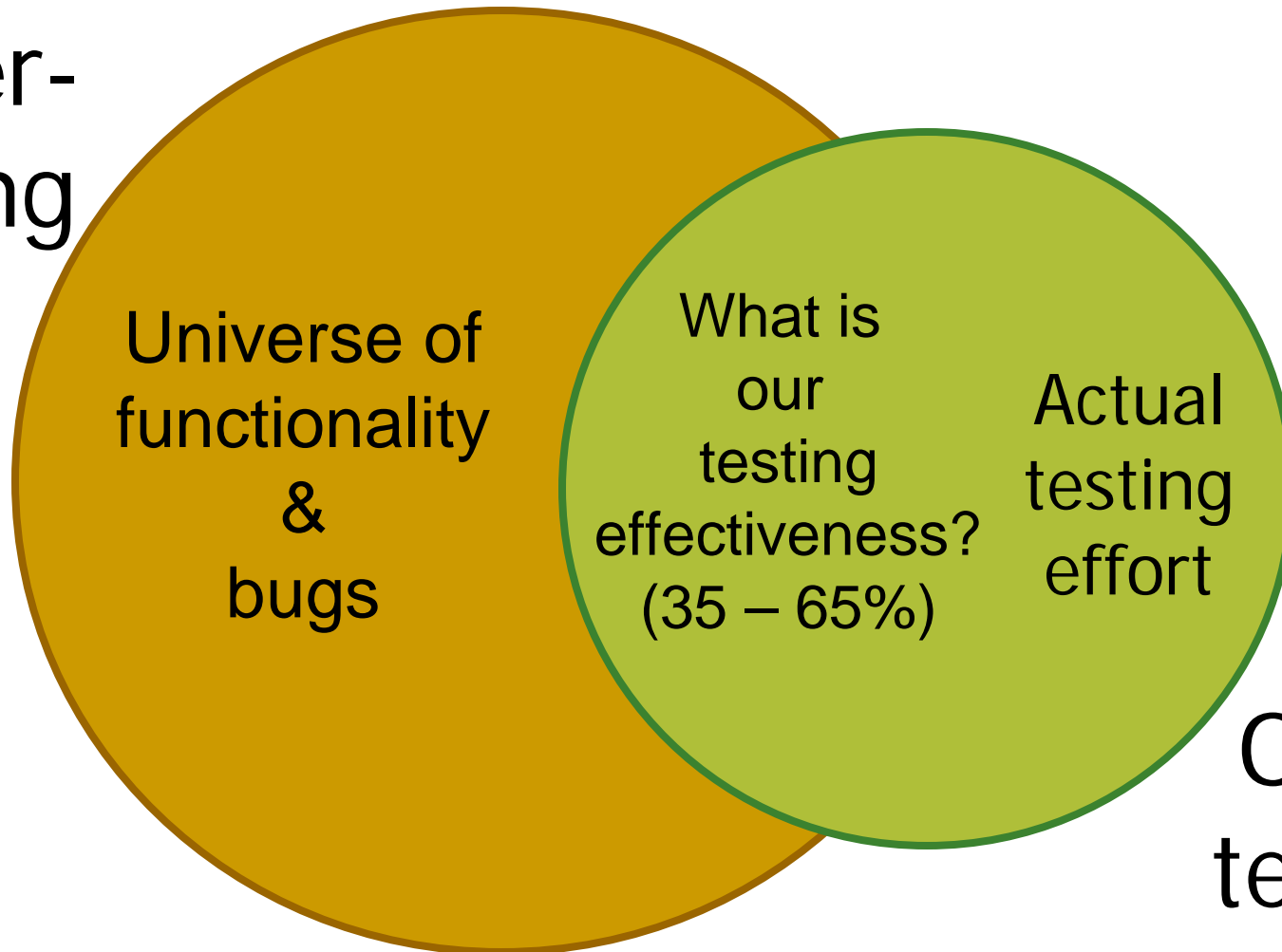


- Anecdotal evidence suggests an overwhelming majority of industry testers have **NOT** read more than one book on software testing
- Less than 10% of testers have formal training in test techniques – Dorothy Graham
- A majority of testers lack sufficient technical skill to write effective test automation, participate in code reviews, or design white box test cases
- A majority view the job as simply finding bugs!

# How effective are our tests?



Under-  
testing



# Microsoft's 2 year case study



- Total participants = 400 untrained testers
- Skill distribution
  - 50% Software Test Engineer (STE) -Tests and critiques software to assure quality and identify potential improvement opportunities.
  - 50% Software Development Engineer in Test (SDET) - Tests and critiques software *components and interfaces in more technical depth, writes test programs to assure quality, and develops test tools to increase effectiveness.*
- Avg. testing experience = 2¼ years
  - 53% – < 1 year experience (20% STE)
  - 12% – 1 - 2 years experience (60% STE)
  - 35% – > 2 years experience (70% STE)



# Weinberg's Triangle Problem



A program reads three integer values. The three values represent the lengths of the sides of a triangle. The program displays a message that states whether the triangle is scalene, isosceles, or equilateral.

**Case Study** – Given 15 minutes, create a set of exploratory test that would adequately\* evaluate the functionality of this triangle program against the functional specification.

Adequate implies test coverage of the critical functionality of the triangle portion of the algorithm



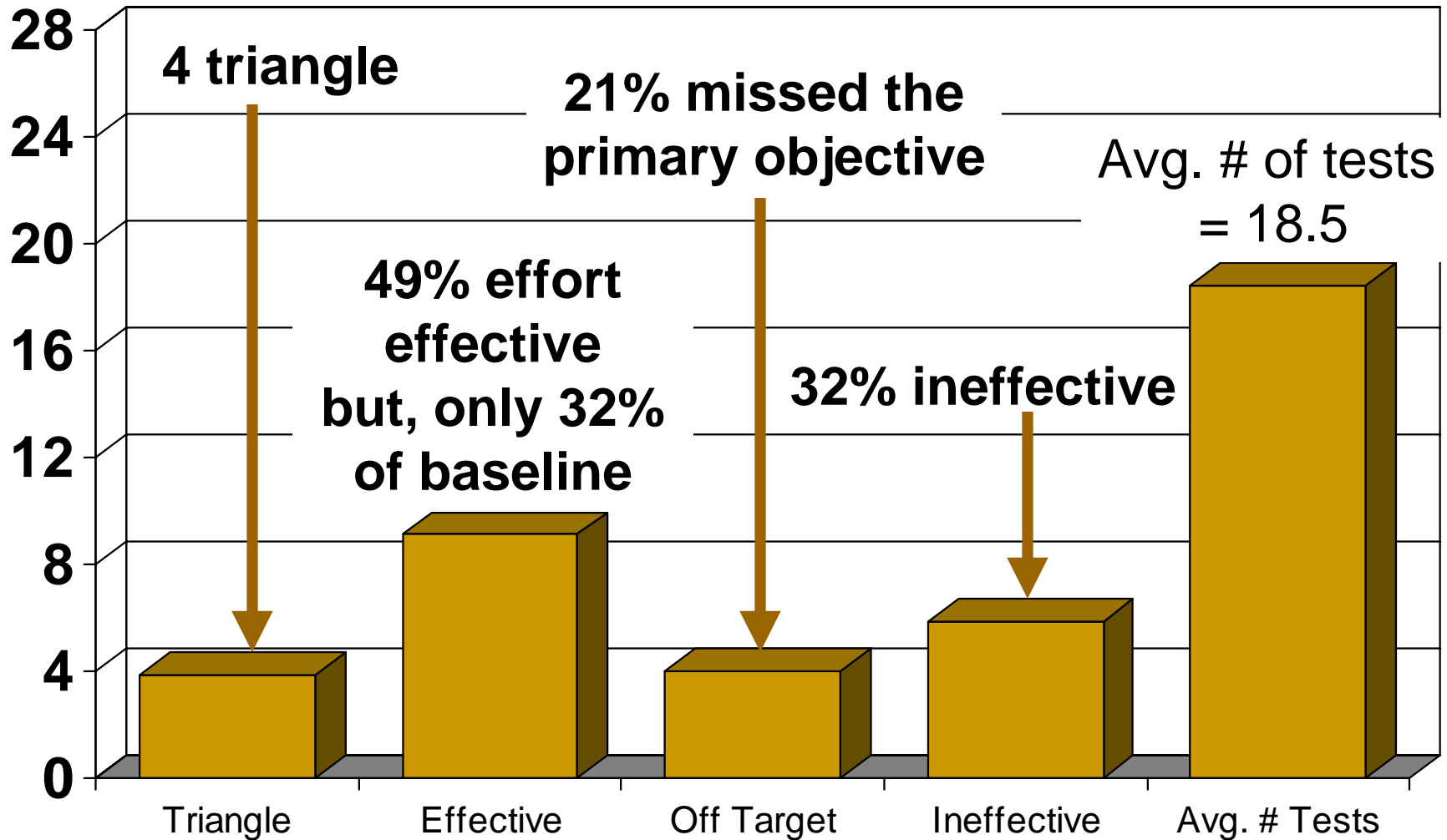
# Evaluating the responses



- Tests grouped into 4 buckets
  - **Triangle** – the tests which specifically targeted the triangle algorithm assuming valid inputs (prev. slide)
  - **Effective** – tests which exercised specific control flow, data flow, operators, or C# methods (*including the Triangle tests*)
  - **Off target** – valid tests usually focused on the user interface, but did not test the triangle algorithm
  - **Ineffective** – tests which would not have proved or disproved anything not covered by a previously executed *effective test (based on our implementation)*

*Baseline - 28 tests based on the algorithm implementation, C# language, invalid inputs, minimum boundary tests, and integer overflow*

# Overall Results



# So, how did we do?



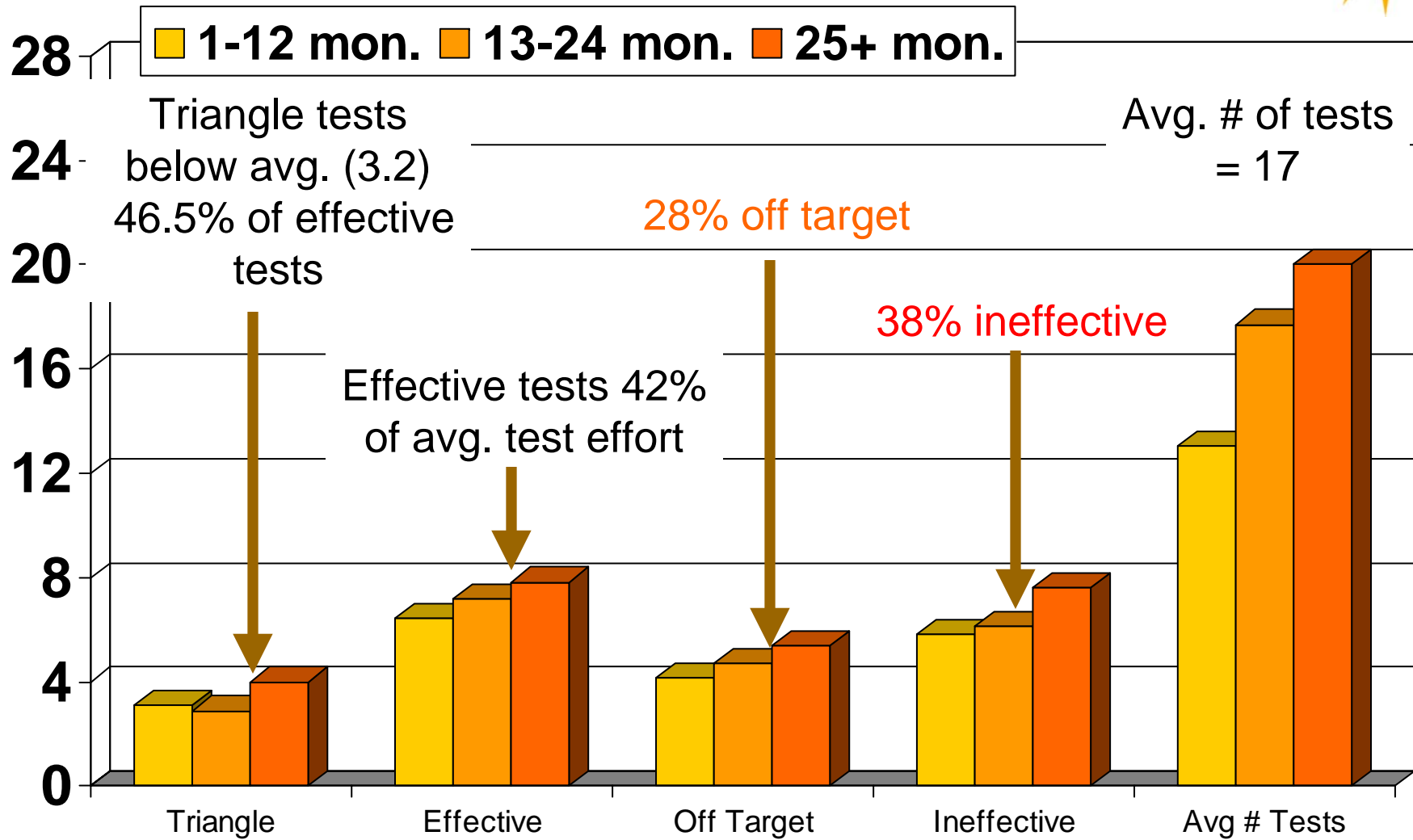
IF ((Side A + Side B <= Side C)  
OR (Side B + Side C <= Side A)  
OR (Side A + Side C <= Side C)) } *84% probability of no testing*  
THEN *input does not equate to valid triangle*

IF (Side A == Side B) AND (Side B == Side C)  
THEN *triangle is equilateral*

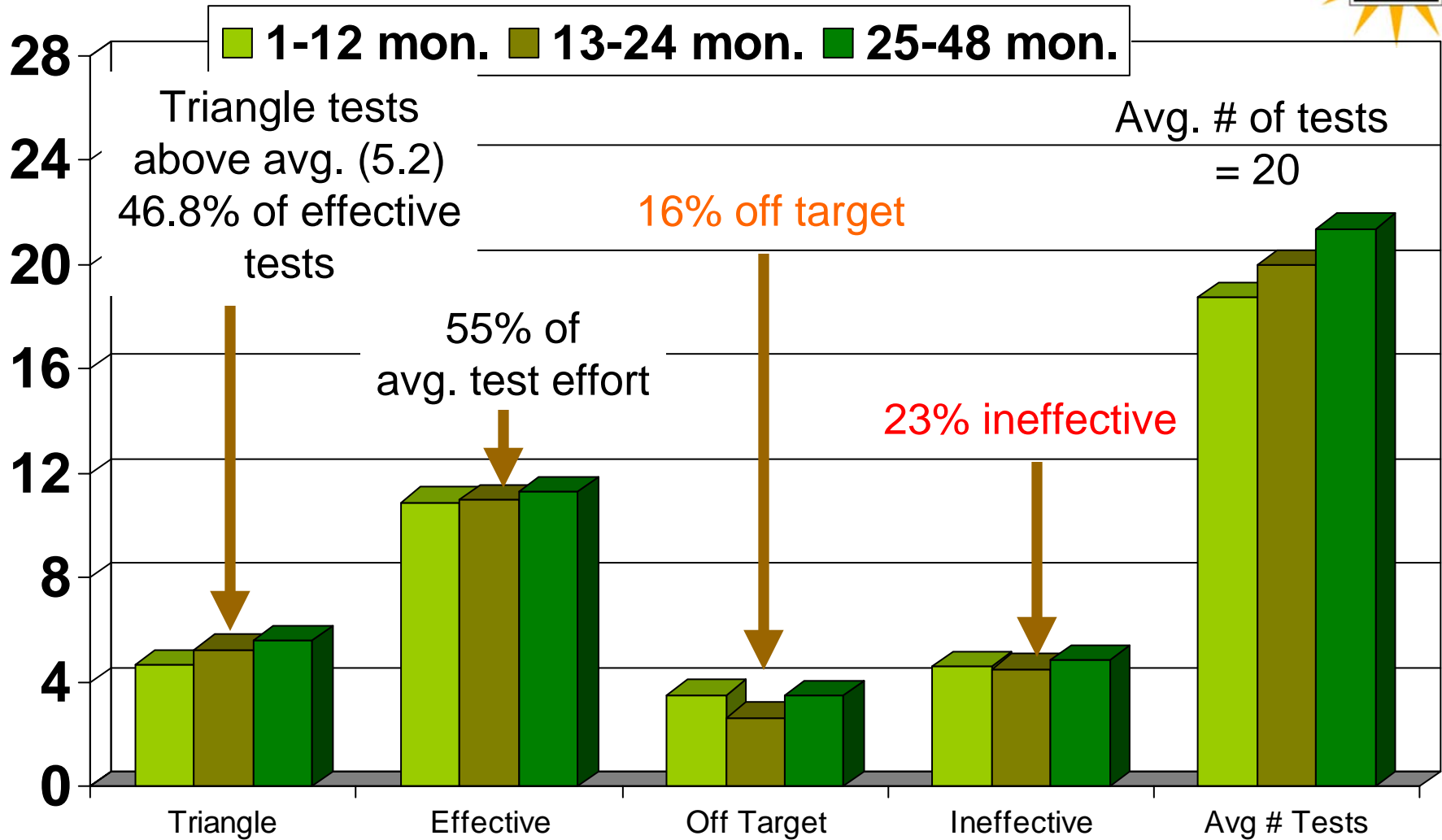
ELSE IF (Side A == Side B)  
OR (Side A == Side C)  
OR (Side B == Side C) } *33% probability of testing*  
*only 1 Isosceles permutations*  
THEN *triangle is isosceles*

ELSE  
*triangle is scalene*      **Avg. of 4 triangle tests =**  
   **36% effectiveness**  
   **(assuming valid inputs 0<> max int)**

# STE Results by Experience



# SDET Results by Experience



# Detailed analysis



- Both STE and SDET approx. 71% valid tests
  - STE 42% of tests were effective + 28% off target
  - SDET 55% of tests were effective + 16% off target
- **Which group better met the objectives in evaluating the primary functionality?**
  - 38% of tests by STEs were ineffective
  - 23% of tests by SDETs were ineffective
- ***Which group of testers were more efficient?***

***Do technical skills in software engineering increase testing effectiveness & efficiency?***

# What's Microsoft doing?



- Microsoft is moving in a different direction
- Refocus building software as an engineering process
- Test engineers (no call center) do more

***Retention***

***Training***

***Hiring***

# Changes at Microsoft- Hiring



## ➤ Testing attributes

- Problem solving, curiosity, adaptability, quality focus, efficiency and effectiveness, thrives on challenges, drive and self-motivation, passion for technology, detail oriented, customer focus, communication skills, independent, growth capacity, etc.
- Domain and system knowledge

## ➤ Engineering background

- Computer science, math, electrical engineering, etc.
- Test engineers who can perform all aspects of software testing
- 16 STE level positions, 565 SDET positions

## ➤ Industry expertise

- Senior experts in the industry
- 84 SDET Lead positions, 7 Test Architect position



# Changes at Microsoft- Training



- Intensive hands-on training (approx. 40 hours)
  - Functional techniques – boundary value analysis, equivalence class partitioning, combinatorial analysis
  - Structural techniques – control and data flow, code coverage analysis
  - Test case design – reusability,
  - Methodologies & tools – exploratory, security, globalization, accessibility, compatibility, etc.
  - Specification reviews and inspections
  - Code review – static analysis, data errors, process
  - Debugging – exceptions, data errors, memory leaks
  - Model based testing – theory and implementation

# Changes at Microsoft- Retention



- Stop the “brain-drain” from our discipline
  - Technical training opportunities to make good testers better test engineers
  - Remove the “glass-ceiling” in promotions
  - Establish senior individual contributor (non-management) roles such as Sr. SDET, Test Architect
  - Greater scope of influence – drive quality upstream
  - Greater challenges in testing
    - Automation - effectiveness, reusability, refactoring
    - Architecture review – testability
    - Risk analysis – churn metrics, dependency analysis
    - Root cause analysis – defect prevention

# So, how do we get better?



## ➤ Increase technical skills

- Programming concepts
- Modern programming language
- Domain and system expertise

## ➤ Increase professional knowledge

- Formal training, university courses, conferences, etc.
- Mentoring, study groups, self-study
- Books, magazines, industry white papers
  - *A Practitioners Guide to Software Test Design* – Lee Copeland
  - *Software Testing Techniques 2<sup>nd</sup> Ed.* – Boris Beizer
  - *The Art of Software Testing 2<sup>nd</sup> Ed.* – Glenford Myers
  - *Testing Object Oriented Systems* – Robert Binder

**“...competition is tougher, you have to redouble your efforts to increase your skill and value.”** B. Marick, *Better Software*, March 2005

# Summary



- Many testers in the industry are under-trained
- Increased technical skills add value
- Test engineers must become more multifaceted
- Software testing is an engineering discipline in the professional field of computer science
  - We employ systematic procedures to accomplish complex tasks; techniques not gimmicks
  - We consistently use our knowledge, experiences and skills to improve our tests and our processes
- **It's all about becoming more professional!**

---

# Questions?



---

**Testing is our profession;  
Quality is our passion!**