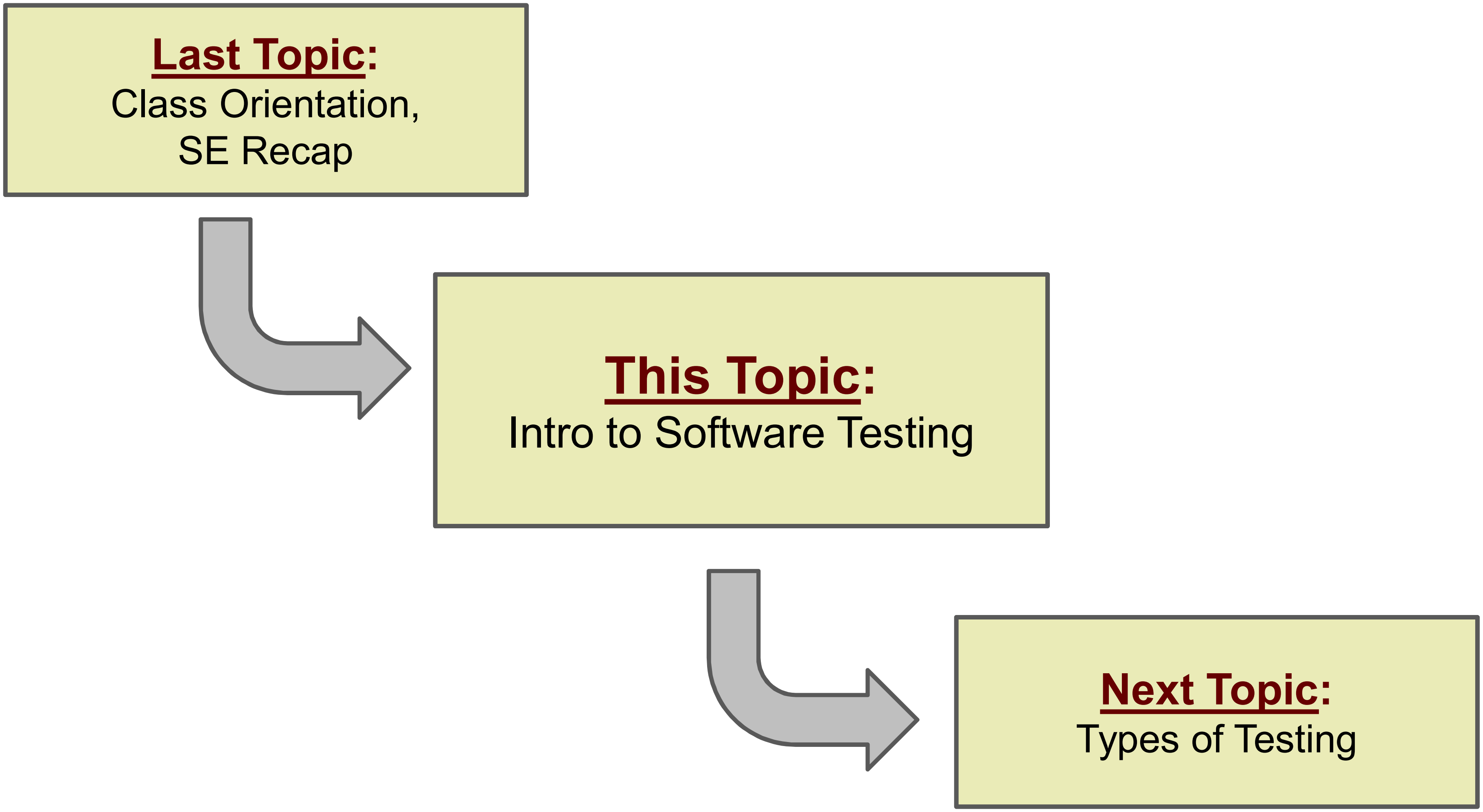
CSE 4283 / 6283
Software Testing and QA

Dr. Tanmay Bhowmik

tbhowmik@cse.msstate.edu

Special thanks to Dr. Nan Niu & Dr. Byron Williams

Agenda

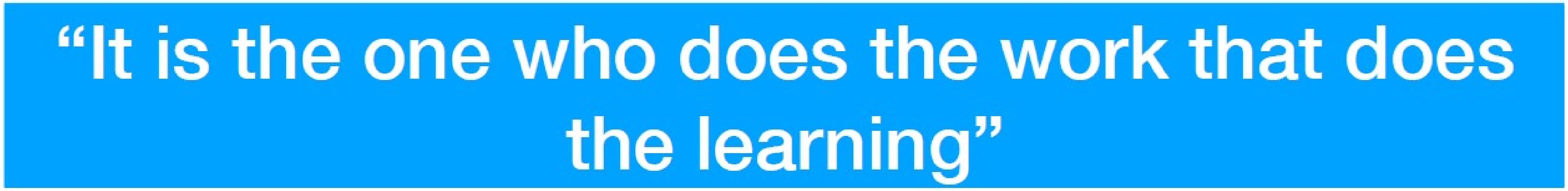


Last Topic:
Class Orientation,
SE Recap

This Topic:
Intro to Software Testing

Next Topic:
Types of Testing

What is the best way to learn?

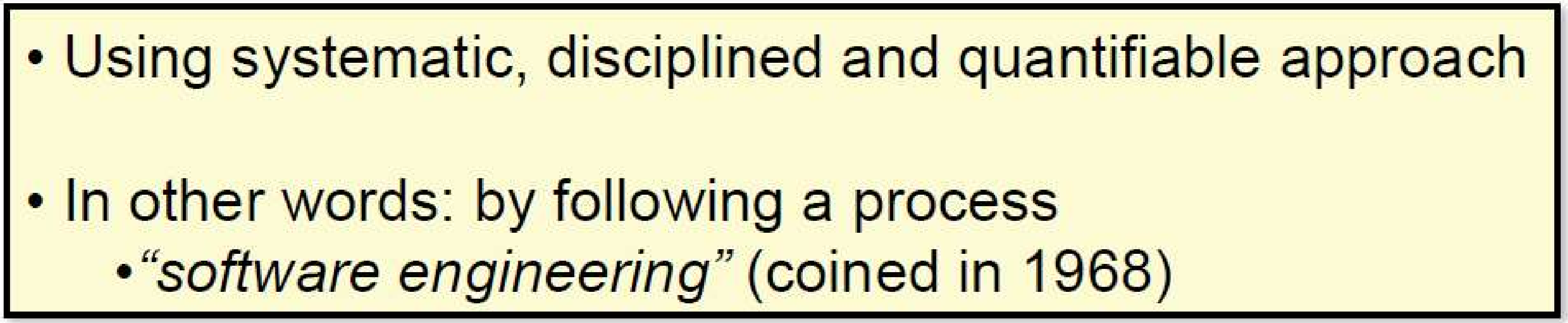


“It is the one who does the work that does the learning”

–Terry Doyle

SE Reviews (3 Questions)

- Q1: Why do we engineer software?
  - – Deliver quality software
- Q2: How do we achieve quality software (product)?

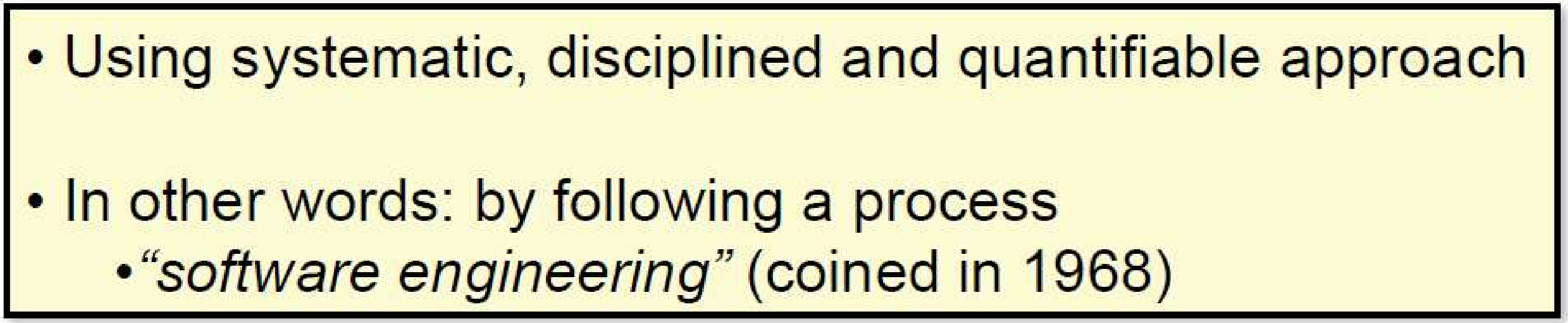


- Using systematic, disciplined and quantifiable approach
- In other words: by following a process
  - *“software engineering”* (coined in 1968)

- Q3: What are process (lifecycle) models?
  - – (What role do “testing & QA” play in these models?)

SE Reviews (3 Questions)

- Q1: Why do we engineer software?
  - Deliver **quality** software
- Q2: How do we achieve quality software (product)?

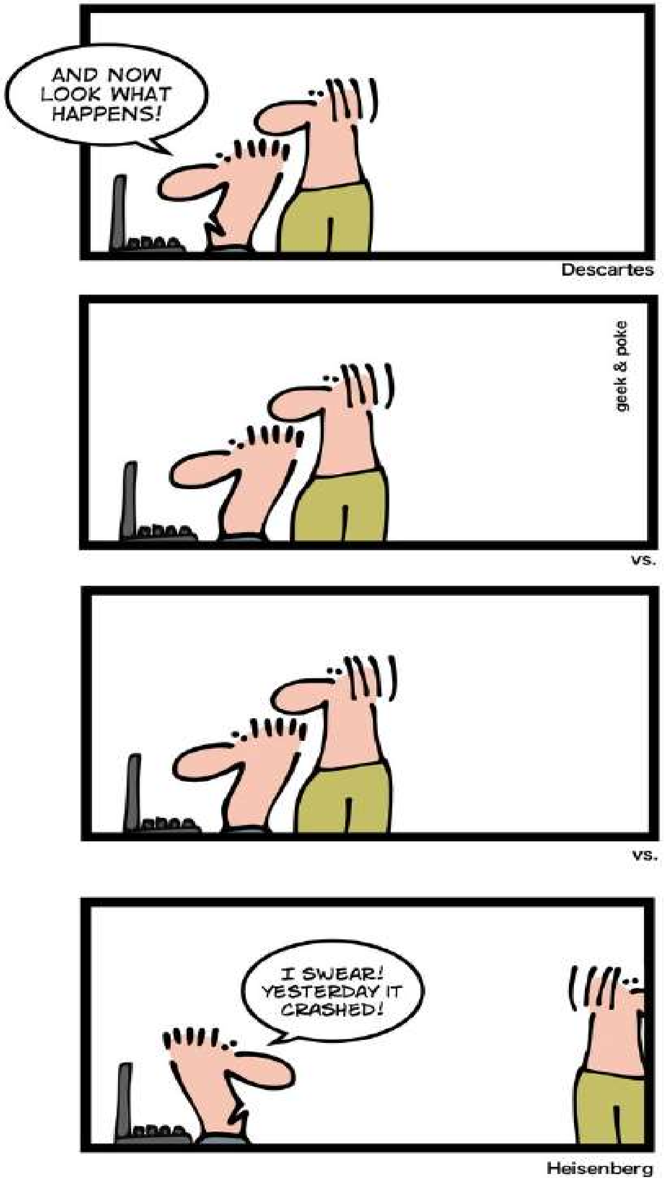


- Using systematic, disciplined and quantifiable approach
- In other words: by following a process
  - *"software engineering"* (coined in 1968)

- Q3: What are process (lifecycle) models?
  - (What role do "testing & QA" play in these models?)

Quality

- What is your definition of high "**quality**"?
- Webster's Dictionary defines "quality" as: ***a degree of excellence***
- *Deliver software system that…*
  - does what it is supposed to do
  - does the things in a desired way
  - show/demonstrate/prove the above two points



Picture source: Internet

Testing: Key Issues

- Why?
  - – Quality demonstration vs. defect detection and reduction
- How?
  - – Execution of software and checking results
    - Must have some executable
- View (what types?)
  - – Functional/external/black-box
  - – Structural/internal/white-box
- Exit (when to stop?)
  - – Coverage vs. usage-based

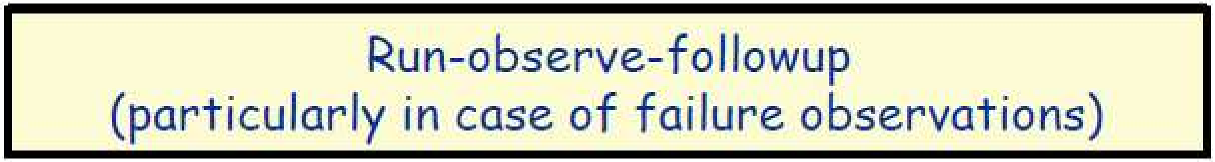
Testing: Why and How?

- Original purpose: demonstration of proper behavior or quality demonstration
  - – "testing" in traditional settings
    - e.g., test the car; test the projector's remote; test the candidate
  - – Evidence of quality or proper behavior

Testing: Why and How?

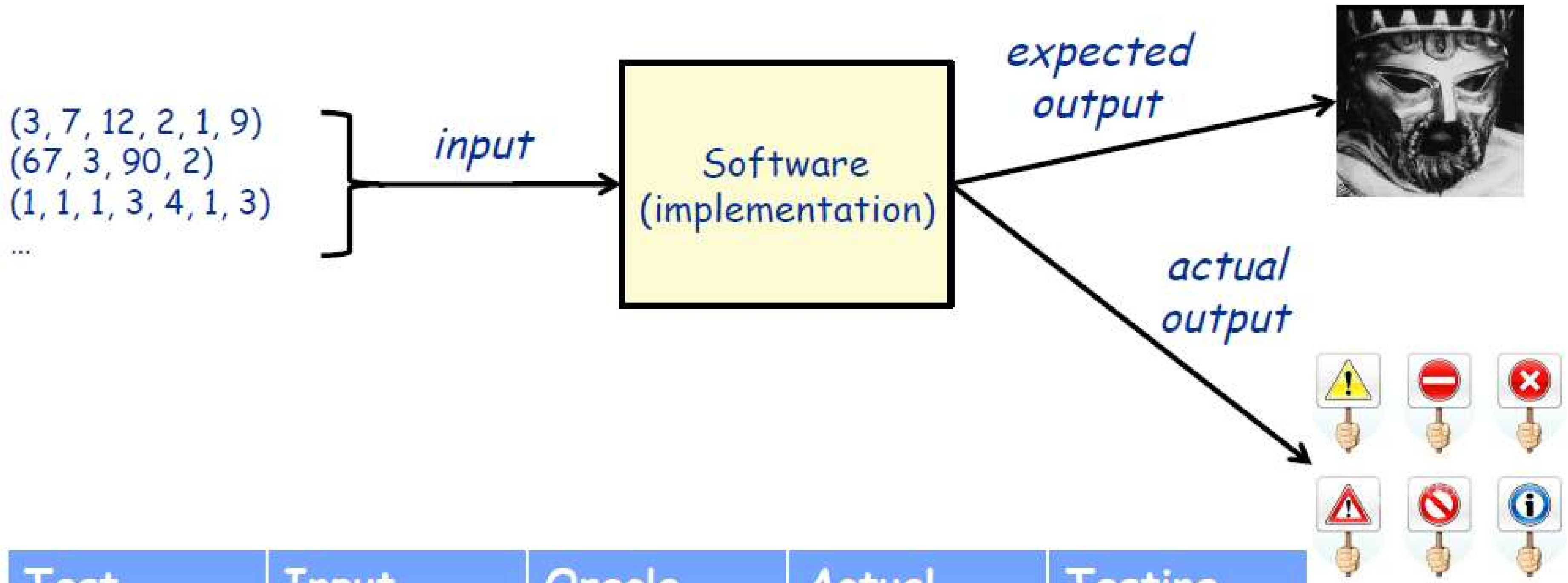
- New purpose: defect detection & reduction
  - Achieve defect-free software
  - Failure observation → fault removal
  - Fault detection → fault fixing
  - Error identification → error prevention

- How?



Run-observe-followup
(particularly in case of failure observations)

“Oracle”: regarded as an infallible authority



(3, 7, 12, 2, 1, 9)
(67, 3, 90, 2)
(1, 1, 1, 3, 4, 1, 3)
...

input

Software
(implementation)

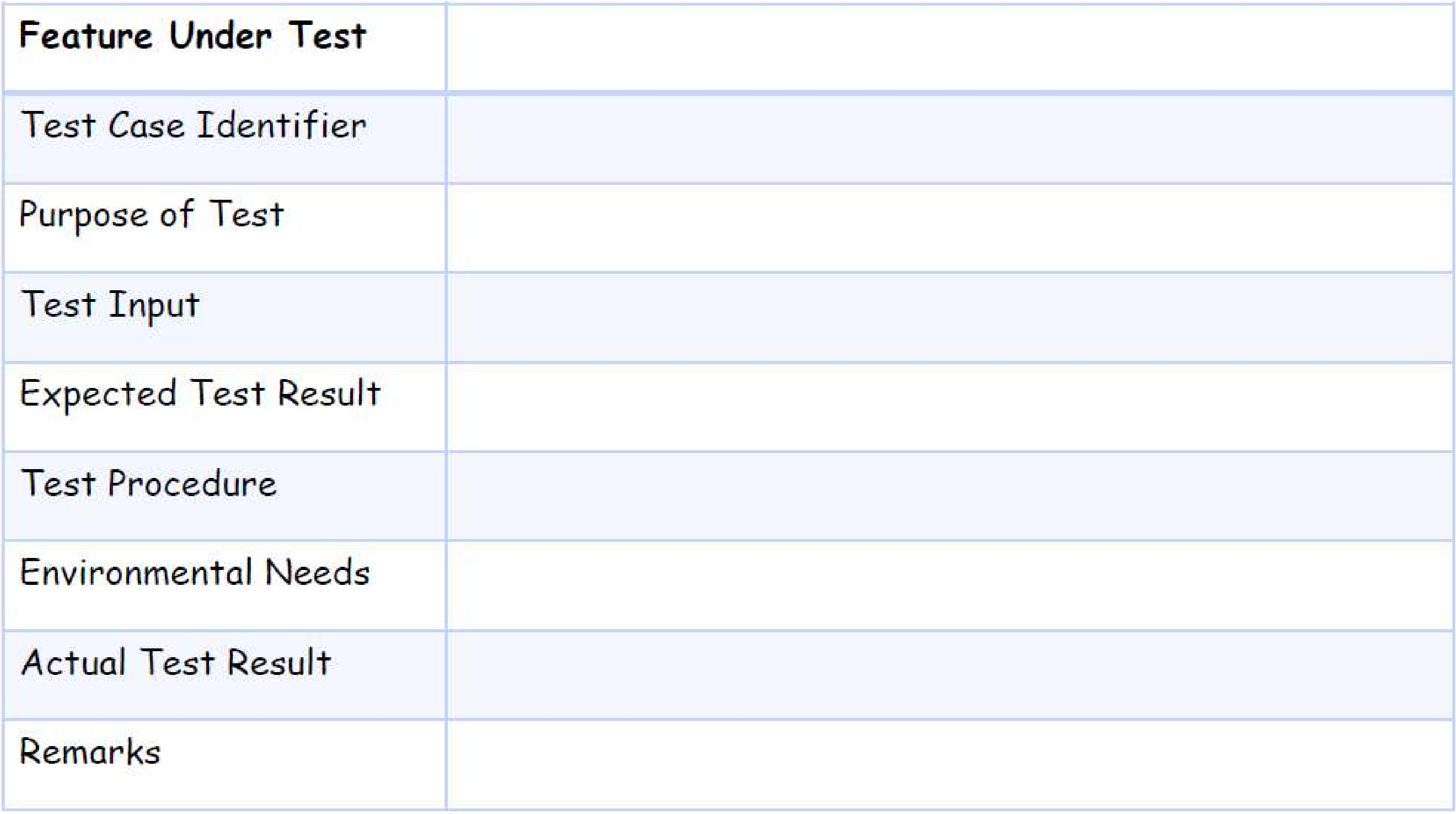
expected output

actual output

Test what?	Input	Oracle (expected output)	Actual output	Testing Result
+	1, 2	3		
+	'A', 'B'	'AB'		
+	1, 'A'	'1A' vs. 66		

Picture source: Internet

Test Case Specification



Feature Under Test	
Test Case Identifier	
Purpose of Test	
Test Input	
Expected Test Result	
Test Procedure	
Environmental Needs	
Actual Test Result	
Remarks	

Test Case Specification: Example

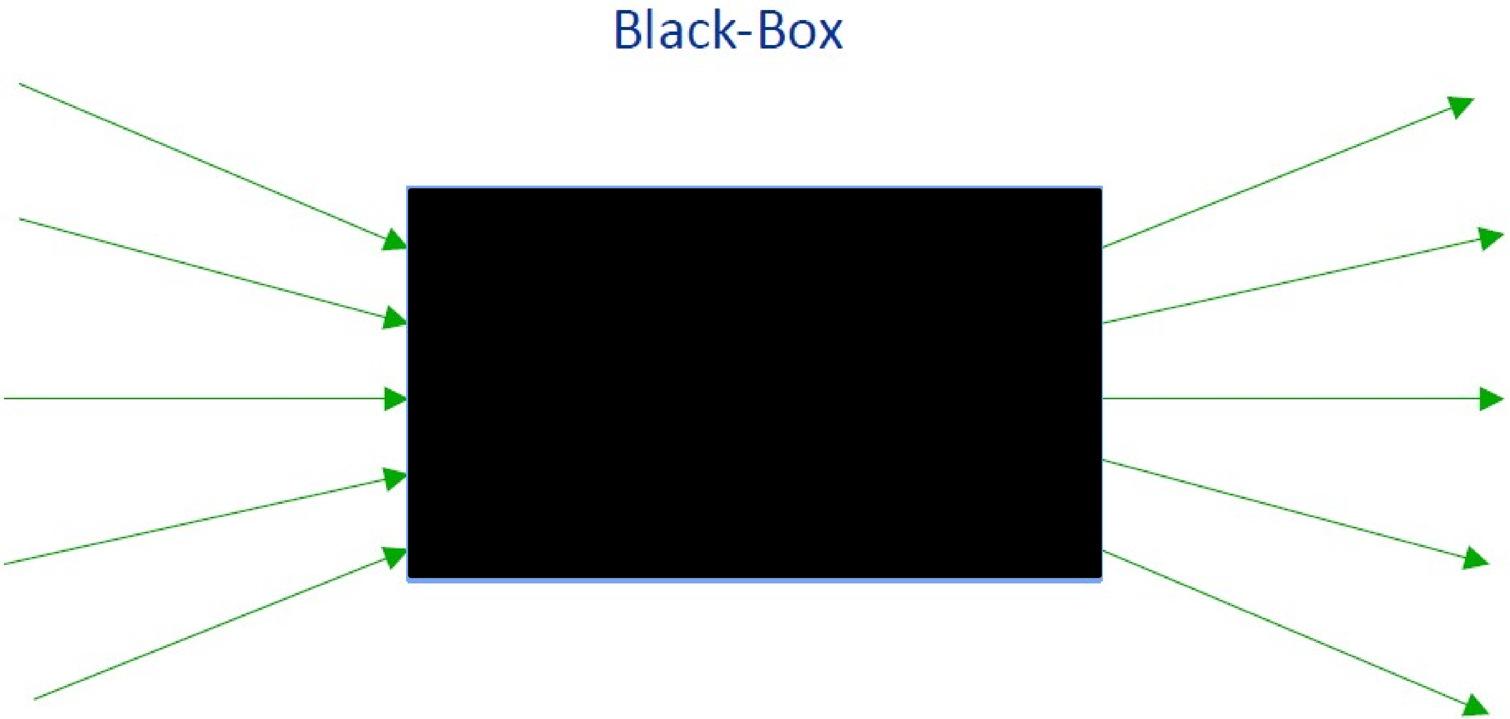
Feature Under Test	**Login**
Test Case Identifier	ICA 01.2
Purpose of Test	This specification describes a test case for testing the "Login" feature. This case has a valid username, but an invalid password.
Test Input	Username: "nn133"; Password: ""
Expected Test Result	System displays the login page with the message: "Invalid username/password".
Test Procedure	Step1: Enter the following into the login page (no quotes): Username: "nn133" Password: "" Step2: Press the "Login" button.
Environmental Needs	FireFox web browser version 2.3 or higher
Actual Test Result	*<to be filled in after test>*
Remarks	*<to be filled in after test>*

Testing: How?

- What artifacts are tested?
  - – Code
- What to test?
  - – from which view?
  - – related: type of faults found?
- When to stop testing?
  - – resource-based vs. quality-based

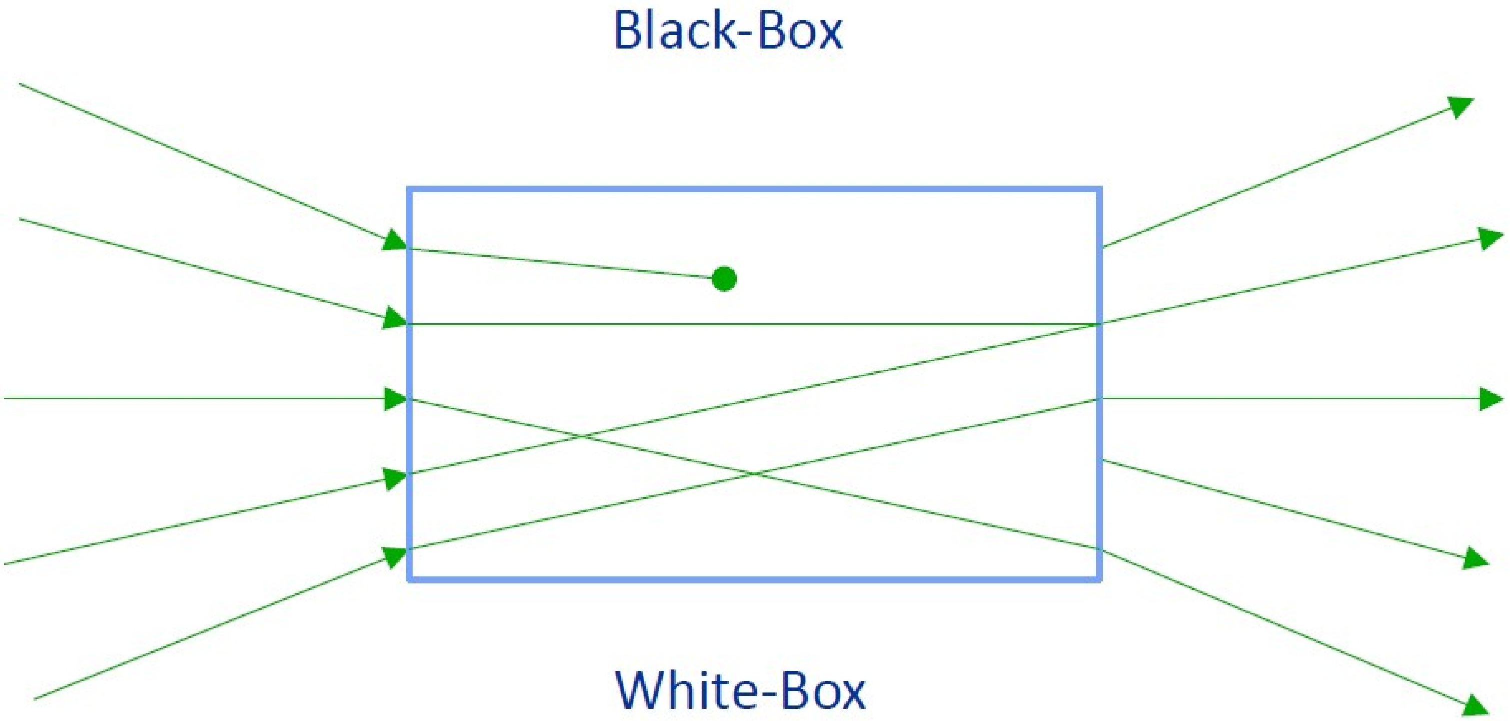
Testing: How?

Black-Box Testing (BBT)
vs. White-Box Testing (WBT)



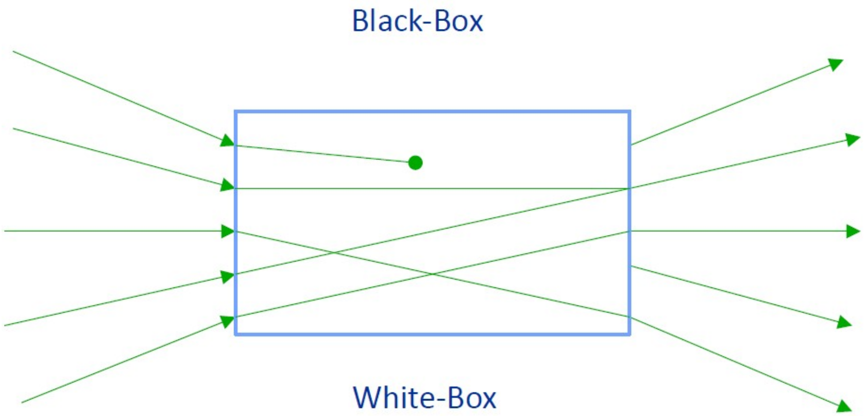
Testing: How?

Black-Box Testing (BBT)
vs. White-Box Testing (WBT)



Functional vs. Structural

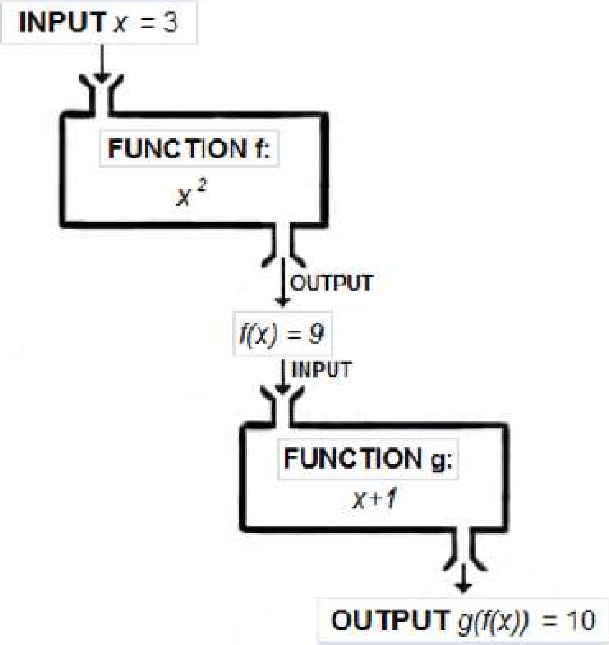
- Functional testing
  - Tests **external** functions
    - as described by external specifications
  - BBT in nature
    - functional mapping: input → output
    - without involving internal knowledge
- Structural testing
  - Tests internal implementations
    - components and structures
  - WBT in nature
    - internal elements visible
    - “white” → seeing through



Functional or BBT

- What is a "function" as in mathematics?

A ***function*** f: $X \rightarrow Y$ is a relation from a domain X to a codomain Y such that for every element x in X, there is a unique y=f(x) in Y.



INPUT $x = 3$

FUNCTION f: x^2

OUTPUT

$f(x) = 9$

INPUT

FUNCTION g: $x+1$

OUTPUT $g(f(x)) = 10$

Relation: $X \rightarrow Y$	Function?	Example
1:1	Yes	f(x)=x+1
m:1	Yes	$f(x)=x^2$
1:n	No	$f(x)=\sqrt{x}$
m:n	No	"friends"

Why does a "function" have to be this way?

Picture source: Internet

Functional Testing

- Function in software
  - functional requirements
  - external/visible features
  - functional programming
  - function/method in a class (OOP)
  - procedure (in C)
  - …
  - Injection, surjection, bijection, partial function, mapping to powerset function and their implications to software testing

Structural or WBT

- Individual modules
  - Statement
  - Path (control flow) testing
  - Data (flow) dependency testing

- Interaction of modules
  - Sub-component
  - Sub-system

Comparing BBT with WBT

	BBT	WBT
Perspective	external behavior (functional)	internal implementation (structural)
Defect Focus	failures	faults
Scale	large software (as a whole)	small objects (looking inside)
Timeline	later (e.g., acceptance testing)	earlier (e.g., unit testing)
Tester	IV&V	developers themselves

Summary

- Testing
  - Why?
    - Quality demonstration vs. defect reduction
  - How?
    - Execute-observe-analyses/followup
  - What types?
    - BBT vs. WBT
  - (When to stop?)

- Next week
  - Testability, Types of Testing

Thank You