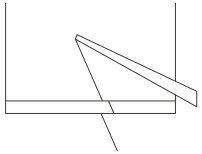
Bring ideas to life
VIA University College



Test & Test Cases

SWE1

What is good quality for Software

Designed Well

- High Cohesion
- Loose Coupling
  - Independent modules
- Clear Interfaces
- Little Redundancy
- Layered design
- Designed for Test
- Extensible
- Portable

What is good quality for Software?

Maintainable

"What makes the difference between working code and great code is maintainability" - David Rachamim

"You don't really know how good someone's code is until you try to change it" - Kristopher Johnson

- Well documented
- Code is readable – stick to coding and naming standards
- Code is simple – KISS Keep it simple stupid!
- Testable
  - Dependency injection
  - Encapsulation
- No "gold plating"
- Only optimised if needed
- Automated tests

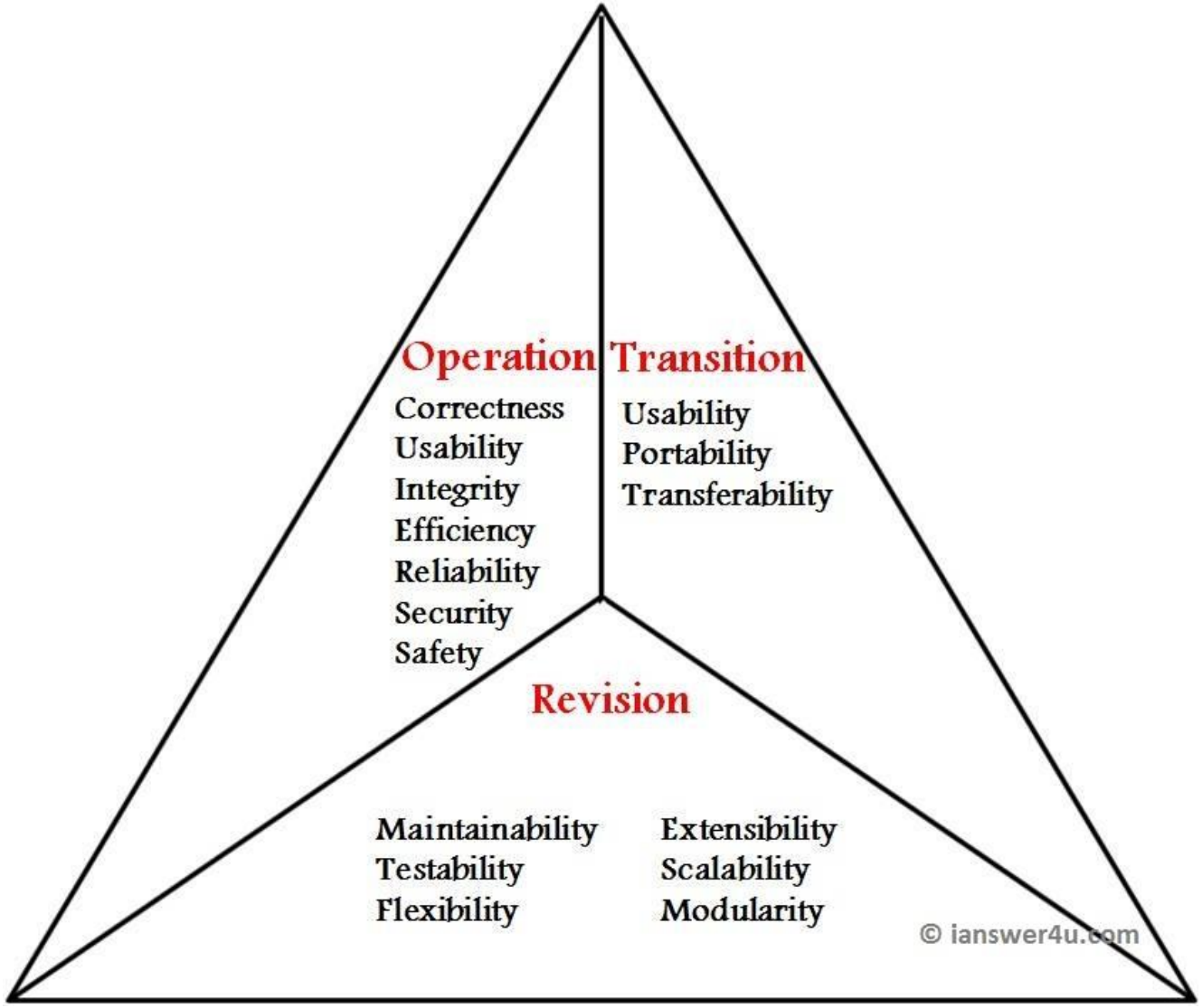
What is good quality for Software?

It Works

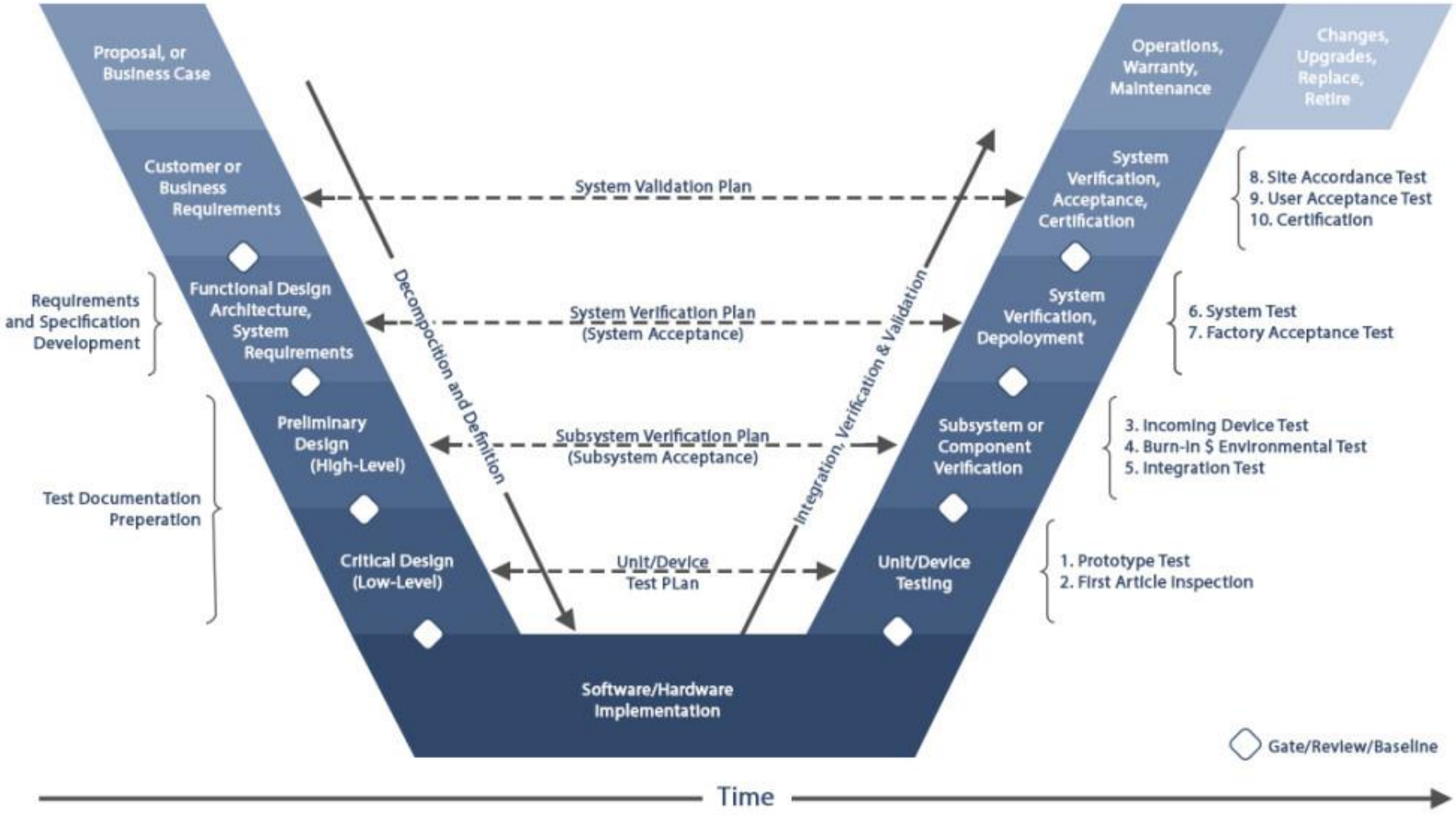
"Beautiful readable , testable, maintainable code that fails to meet the customer needs is still a failure" - codingdave

- As specified by customer
- It solves the intended problems
- Users wants to use it
- It's stable

Good Software Quality



V-Model & Test


Proposal, or Business Case
Customer or Business Requirements
Functional Design Architecture, System Requirements
Preliminary Design (High-Level)
Critical Design (Low-Level)
Software/Hardware Implementation
Unit/Device Testing
Subsystem or Component Verification
System Verification, Depoloyment
System Verification, Acceptance, Certification
Operations, Warranty, Maintenance
Changes, Upgrades, Replace, Retire
System Validation Plan
System Verification Plan (System Acceptance)
Subsystem Verification Plan (Subsystem Acceptance)
Unit/Device Test PLan
Decompcition and Definition
Integration, Verification & Validation
Requirements and Specification Development
Test Documentation Preperation
8. Site Accordance Test
9. User Acceptance Test
10. Certification
6. System Test
7. Factory Acceptance Test
3. Incoming Device Test
4. Burn-in $ Environmental Test
5. Integration Test
1. Prototype Test
2. First Article Inspection
Gate/Review/Baseline
Time

Kinds of tests

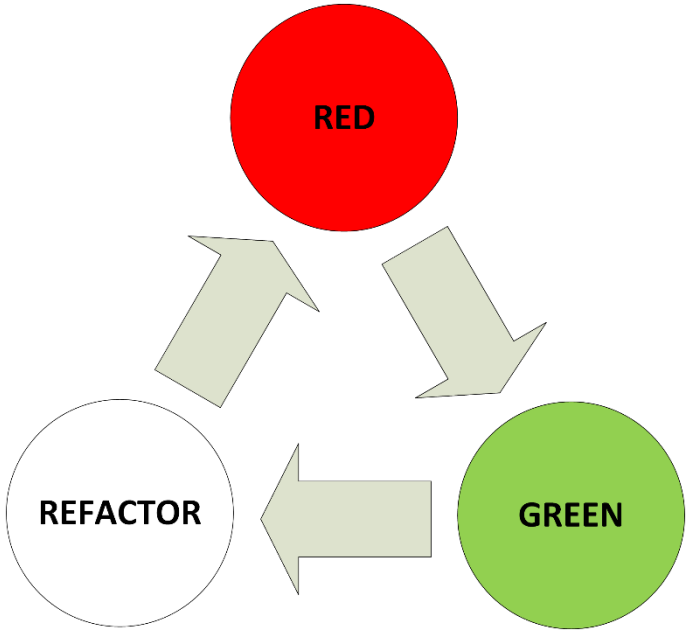
Black box testing
- System Tests
- Acceptance Tests

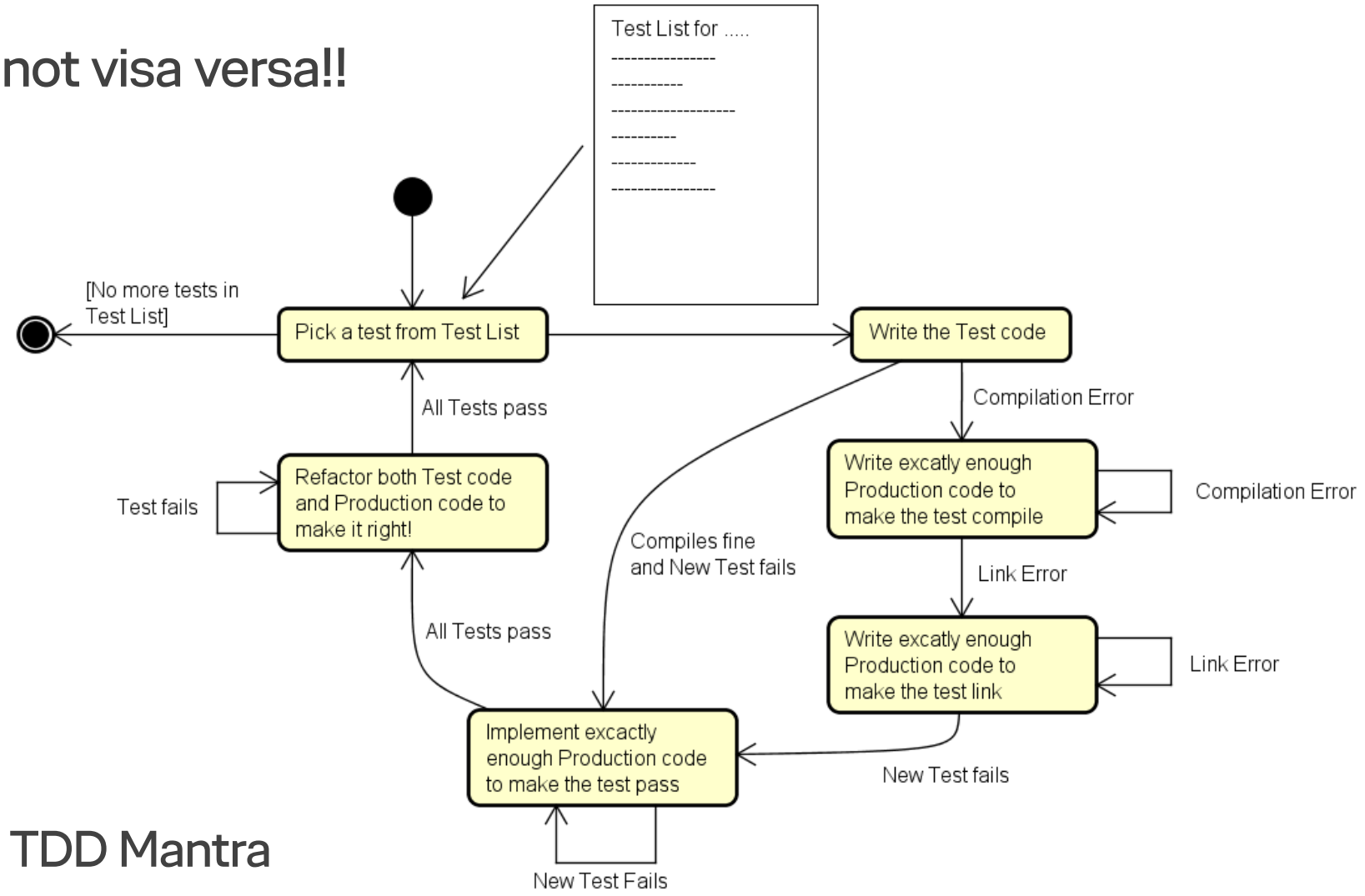
Test Driven Development (TDD)

White box testing
- Unit Tests
- ZOMBIES

What is Test Driven Development (TDD)?

Let code follow test – not visa versa!!





RED-GREEN-REFACTOR – the TDD Mantra

ZOMBIES Guide to Test Cases

ZOMBIES Spelled out:

- Z – Zero
- O – One
- M – Many (or More complex)
- B – Boundary Behaviors
- I – Interface definition
- E – Exercise Exceptional behavior
- S – Simple Scenarios, Simple Solutions

Unpronounceable acronym:
DTSTTCPW. Spelling it out:

Do The Simplest Thing That Could Possible Work.

- Kent Beck

See James W. Grennings explanations:
http://blog.wingman-sw.com/archives/677#more-677

ZOMBIES Guide to Test Cases

Z – Zero

The first test **S**cenarios are for **S**imple post-conditions of a just created object/module.

These are the **Z**ero cases.

ZOMBIES Guide to Test Cases

O – One

The test **S**cenarios are for **S**imple tests dealing with single items

These are the **O**ne cases.

ZOMBIES Guide to Test Cases

M – More or More complex

The test **S**cenarios are for **S**imple tests dealing with more items or more complex scenarios

These are the **M**ore cases.

ZOMBIES Guide to Test Cases

B – Boundary Behaviors

The test **S**cenarios are for **S**imple tests dealing with the boundaries

These are the **B**oundary cases.

ZOMBIES Guide to Test Cases

I – Interface definition

These are not tests, but writing tests defines the needed interfaces for our modules

ZOMBIES Guide to Test Cases

E – Exercise Exceptional behavior

Test all odd situations and be sure that your system can handle them in a defined way

ZOMBIES Guide to Test Cases

S – Simple Scenarios, Simple Solutions

Test simple scenarios one by one, do not test many things in one test case!

Implement the simplest solutions to pass the tests!

TDD Guided by ZOMBIES

Zero

One

Many

Simple Scenarios

Simple Solutions

B Boundaries Behaviors

I Interfaces

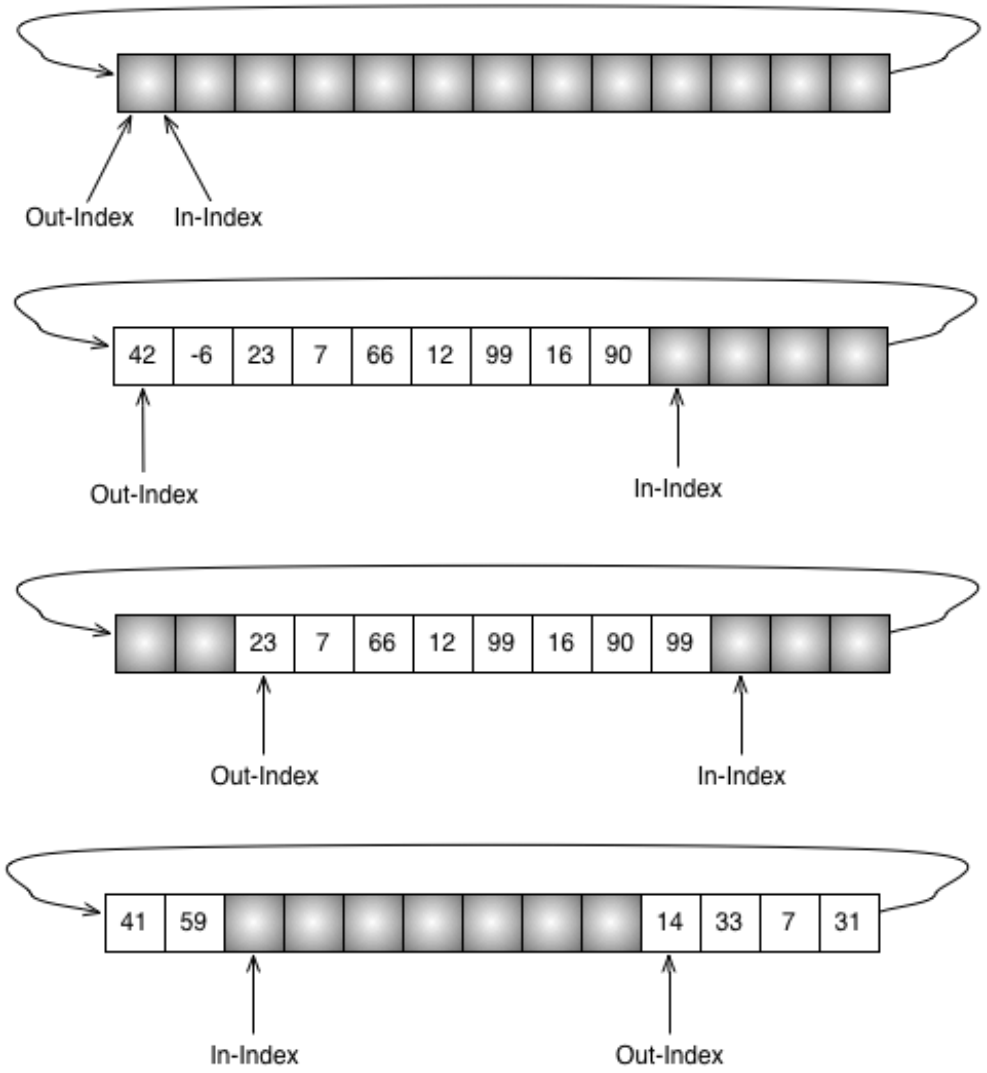
E Exceptions Exercise

Zombies photo thanks to Joel Friesen

A FIFO buffer (Circular Buffer) as example

First understand the problem

- Do some sketches on paper!

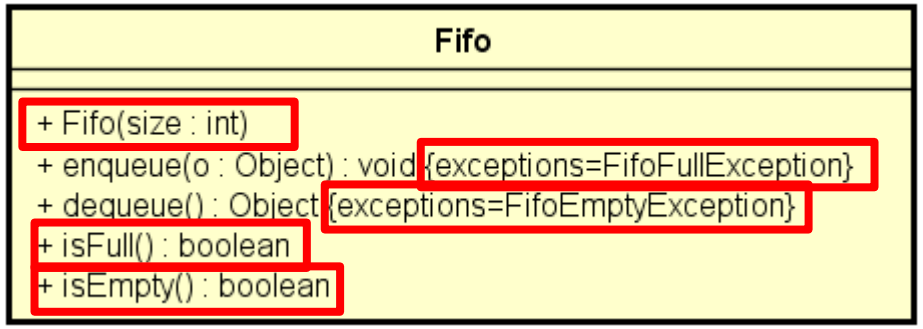


James W. Grenning

TDD Demo

Production- and Test code will be uploaded later!

This is what we ended with:



From the 'I' in ZOMBIES