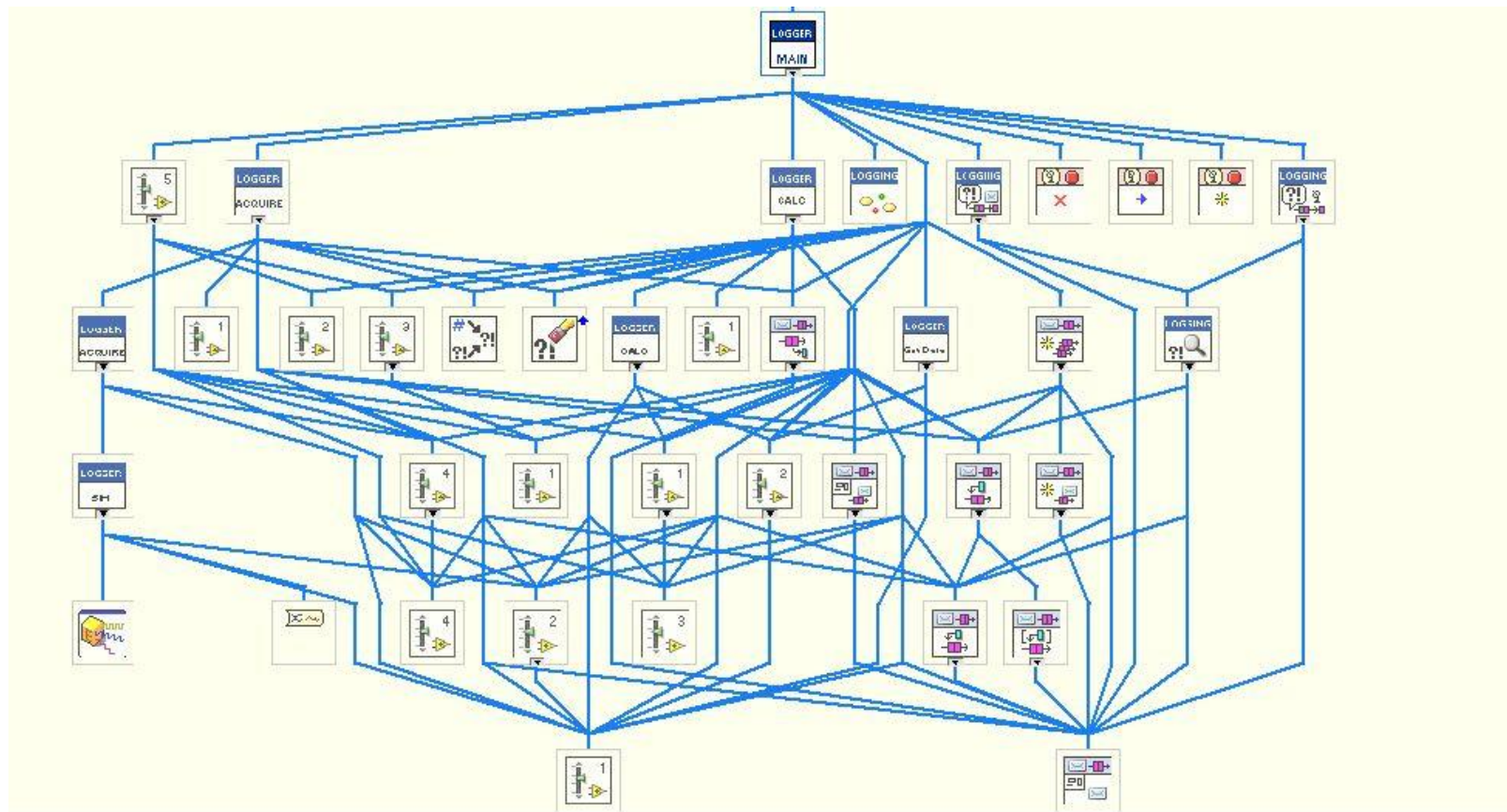


C.A.G

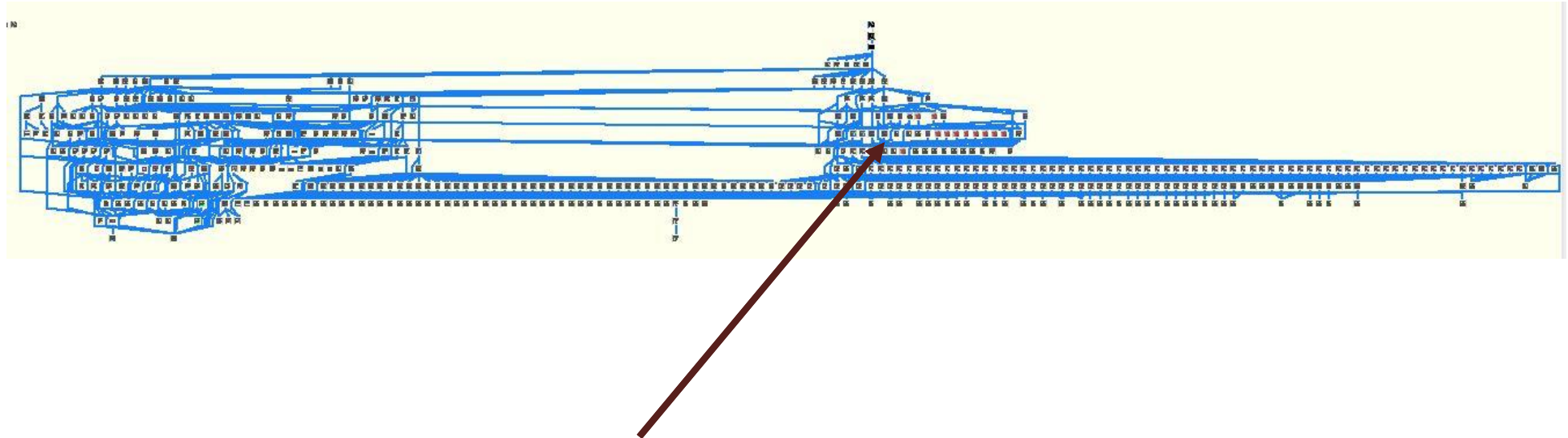
## *Testa din LabVIEW applikation*

Fredrik Edling, Tekn. Dr.

C.A.G



C.A.G



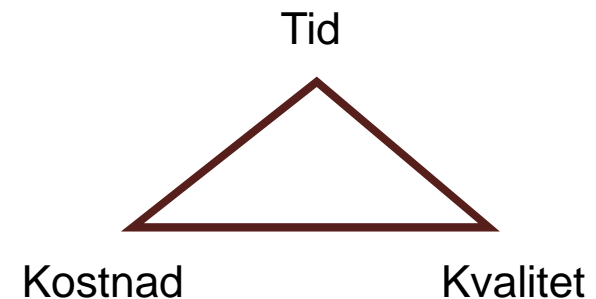
VI:n som du har fixat ett fel i

C.A.G



# När har jag testat färdigt?

- ” Det finns alltid ett testfall till
  - . Ett testfall räcker för att visa att koden inte fungerar
- ” Riskbaserad test
  - . Basera testmängden och testkvaliteten på vilken risknivå som är acceptabel
  - . Prioritera testfallen utifrån riskerna



# Test i olika stadier

## Enhetstest

- Verifiera funktion hos en enhet
- Avser en minsta testbar kodenhet
- Kan vara en modul, eller ett kluster av sammanhörande moduler

## Integrationstest

- Verifiera att enheterna kan interagera med varandra
- Bygga samman systemets enheter, vanligen på delsystemnivå

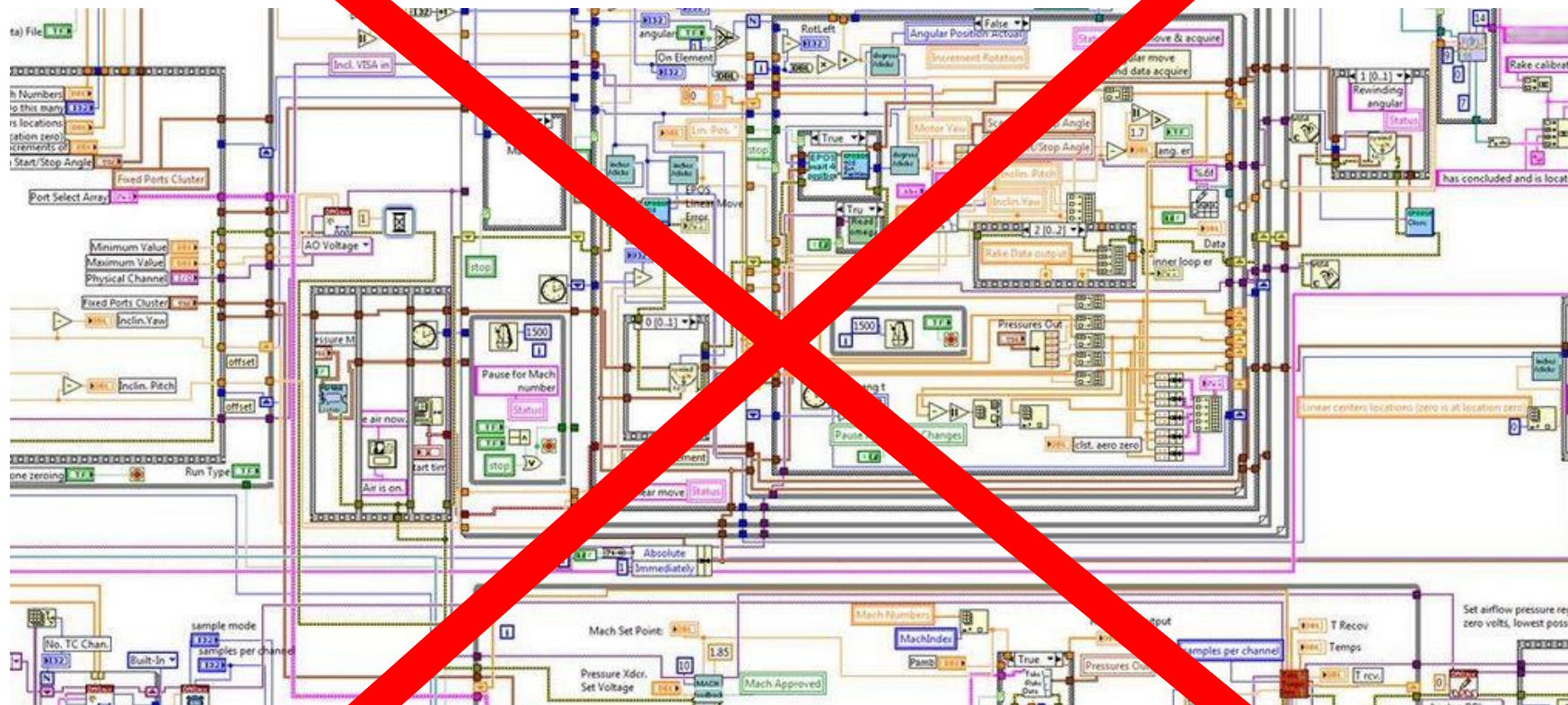
## Systemtest

- Verifiera systemet mot kraven

# Verktyg som underlättar

- ” Statiska tester
  - . VI analyzer
- ” Enhetstest
  - . National Instruments Unit Test Framework
  - . VI Tester från JKI
- ” Systemtest
  - . Teststand
  - . Egna testramverk
- ” Automatiska tester
  - . Jenkins för att styra Continuous Integration
- ” Användargränssnitt
  - . Capture-replay verktyg

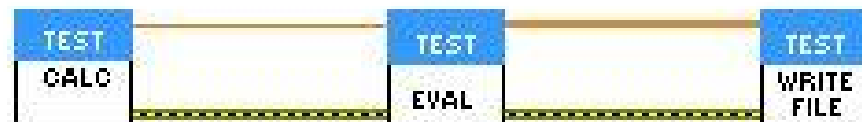
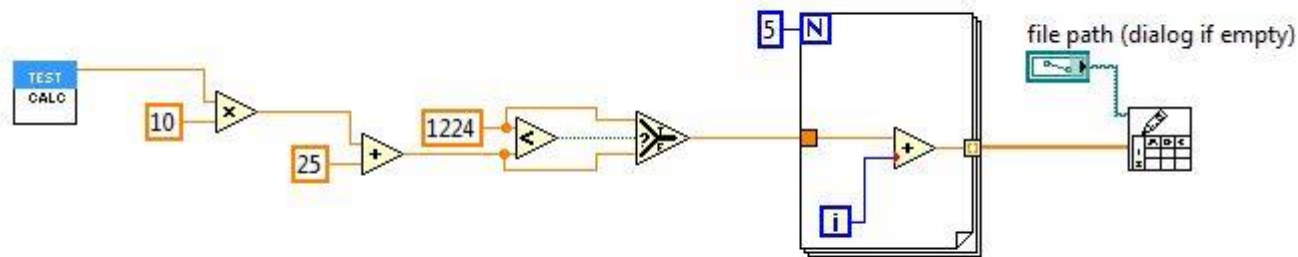




C.A.G



# Modularisering och gränssnitt



# Enhetstester

## NI Unit Test Framework

- Direkt tillgänglig i projektet
- Använd testvektorer för att testa flera värden
- Möjlighet att skriva egna wrappers för testet
- Möjlighet att använda setup och teardown  
vi:s.

## VI Tester från JKI

- Kan testa klasser

# Unit Test Framework

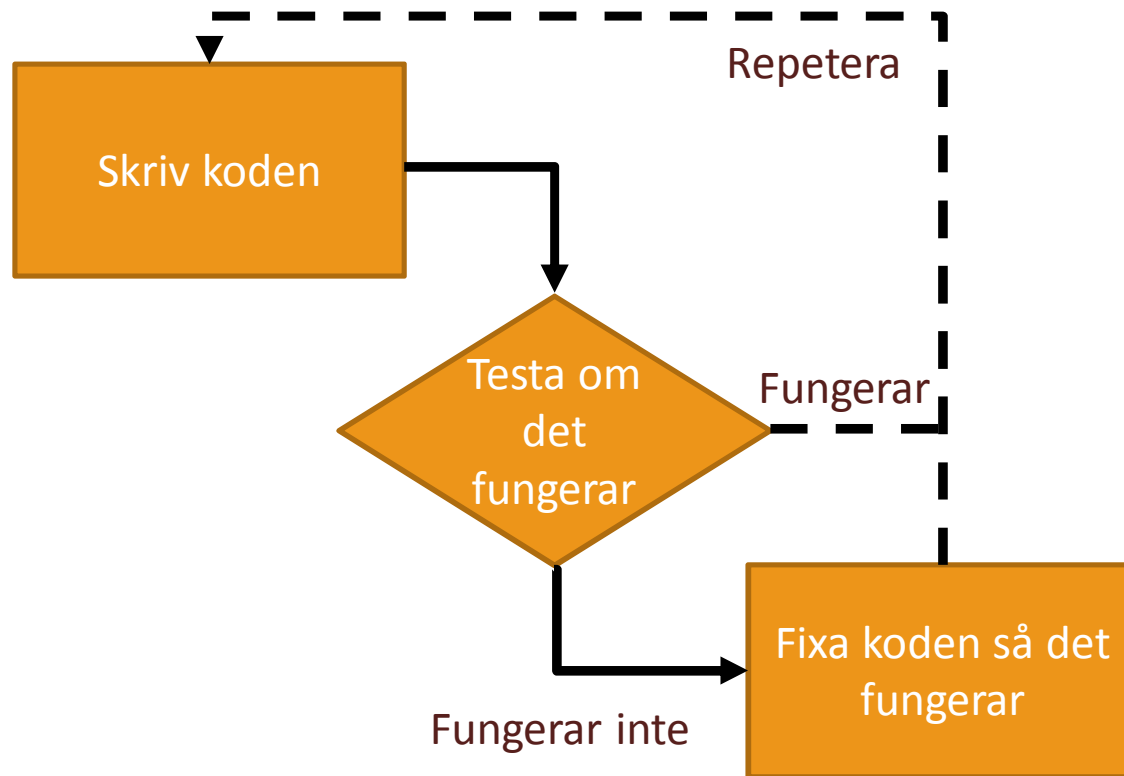
## ” Krav

- . Req1: Om  $X > 10$  multiplicera med 1,5 och subtrahera med 0,5
- . Req2: Om  $X \leq 10$  multiplicera med 0,5

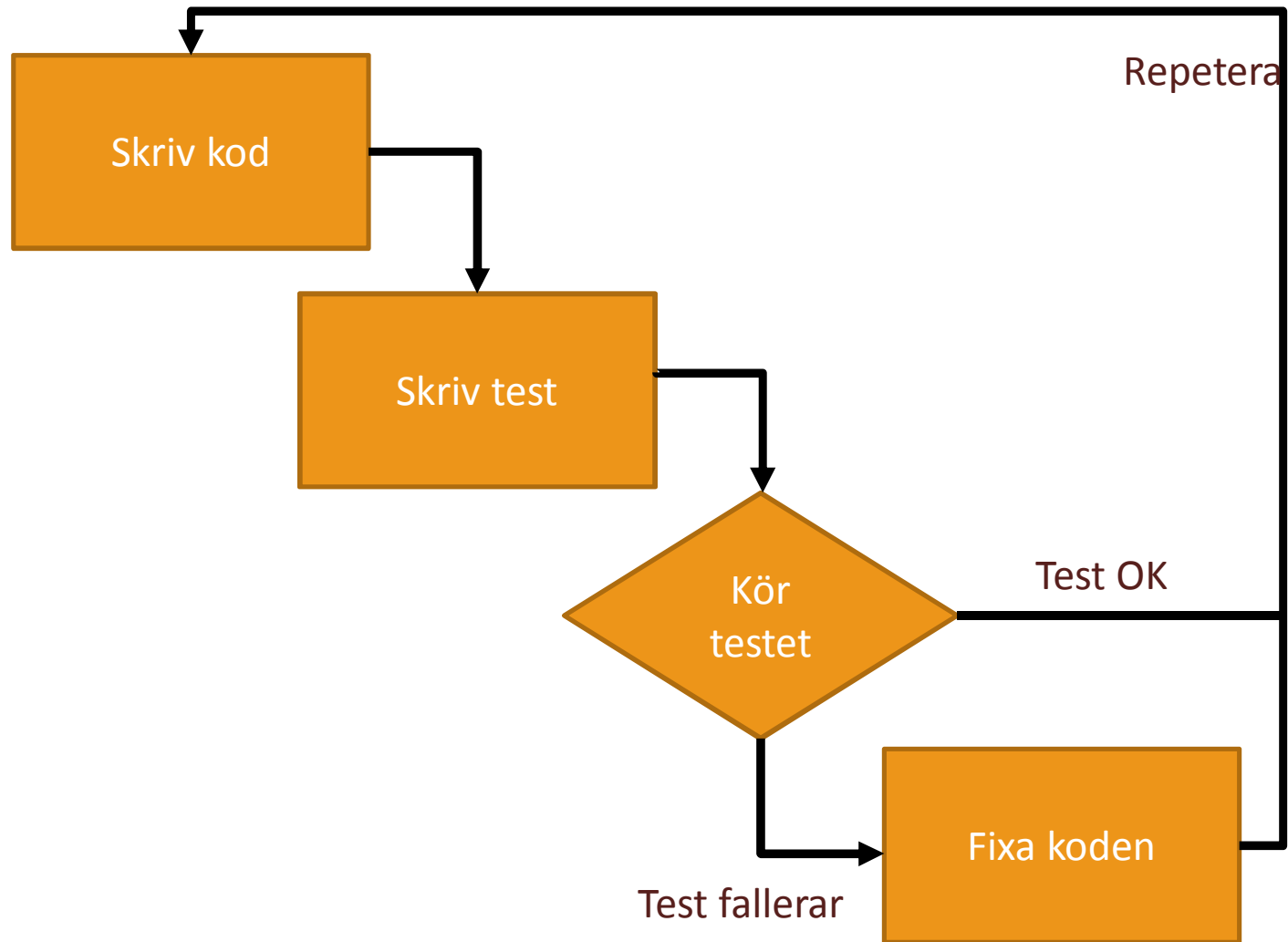
## ” Enkelt gränsvärdestest

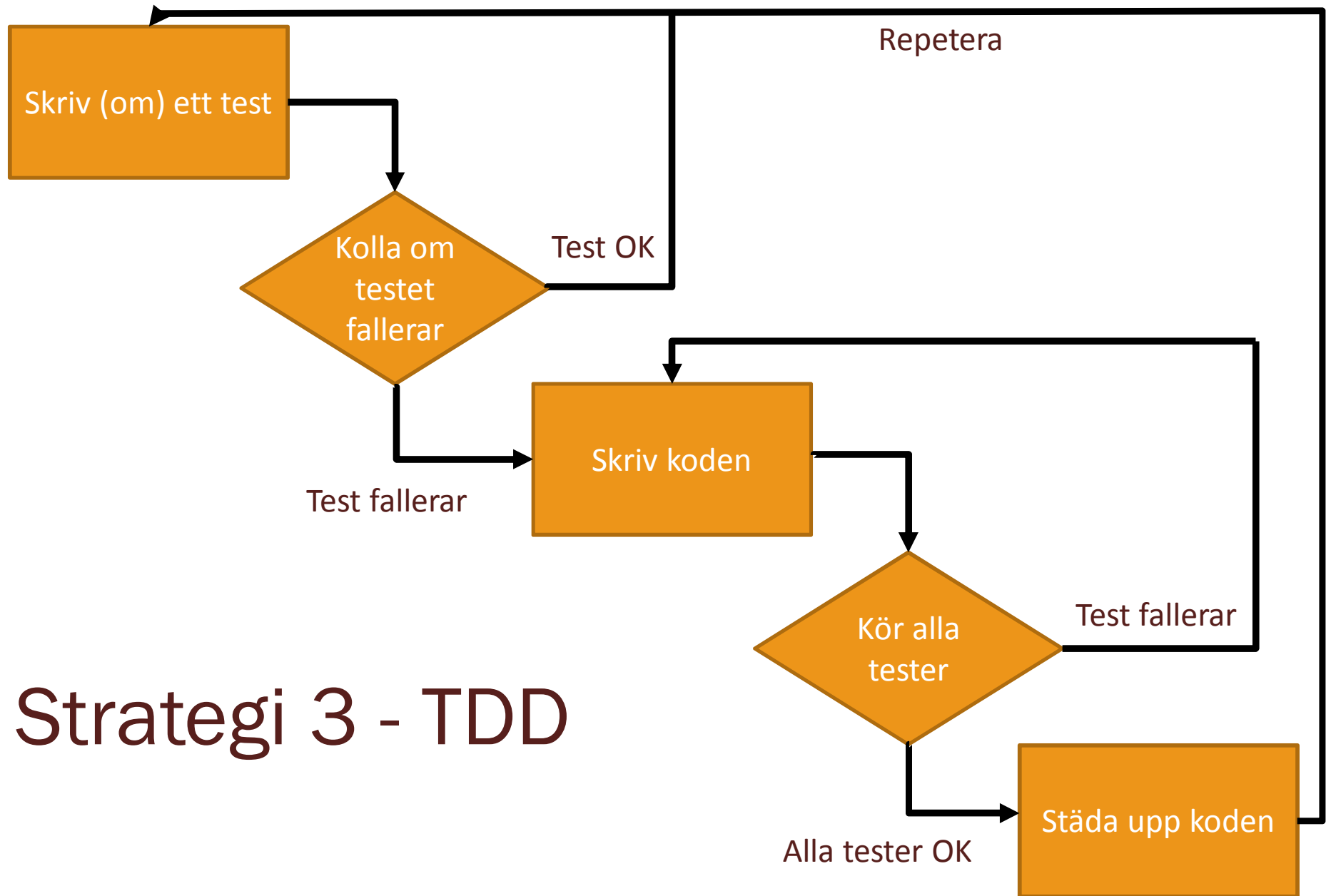
Testfall	Inparameter	Förväntat resultat
TC1	9	4,5
TC2	10	5
TC3	11	16

# Strategi 1



# Strategi 2





## Strategi 3 - TDD

# TDD – testdriven utveckling

## Fördelar

- Fokus på kraven
- Tvingar fram modularisering och undviker onödiga kopplingar mellan moduler
- Mindre risk att onödig kod implementeras



# TDD – testdriven utveckling

## Att se upp med

- Passar inte för test av användargränssnitt och access till exempel till databaser.
  - -> försök separera kod från användargränssnittet
- Dåligt skriva testfall kan orsaka merarbete
- Integrations- och systemtest behöver fortfarande göras



C.A.G

# Continuous Integration

Privat bygge  
Kompilera källkod  
Enhetstester

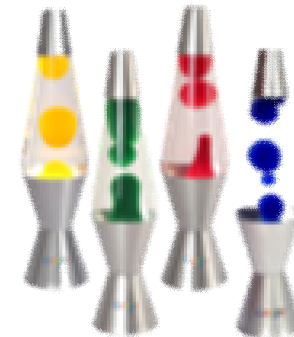
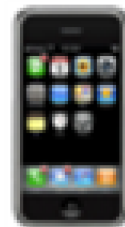
IDE



IDE



Återkoppling



Repository



CI Server



Incheckat bygge  
Kompilera källkod  
Bygga exe-fil  
Enhetstester  
Statiska tester  
Integrationstester

Håll kort: < 10 min

C.A.S

ÖKANDE FÖRTROENDE FÖR BYGGETS FÄRDIGHET

TESTMILJÖN BLIR ALLTMER PRODUKTIONSLIK

SNABBARE ÅTERKOPPLING

Privat bygge

Incheckat bygge

Funktionella tester

Icke-funktionella tester

Tester i olika produktionsmiljöer

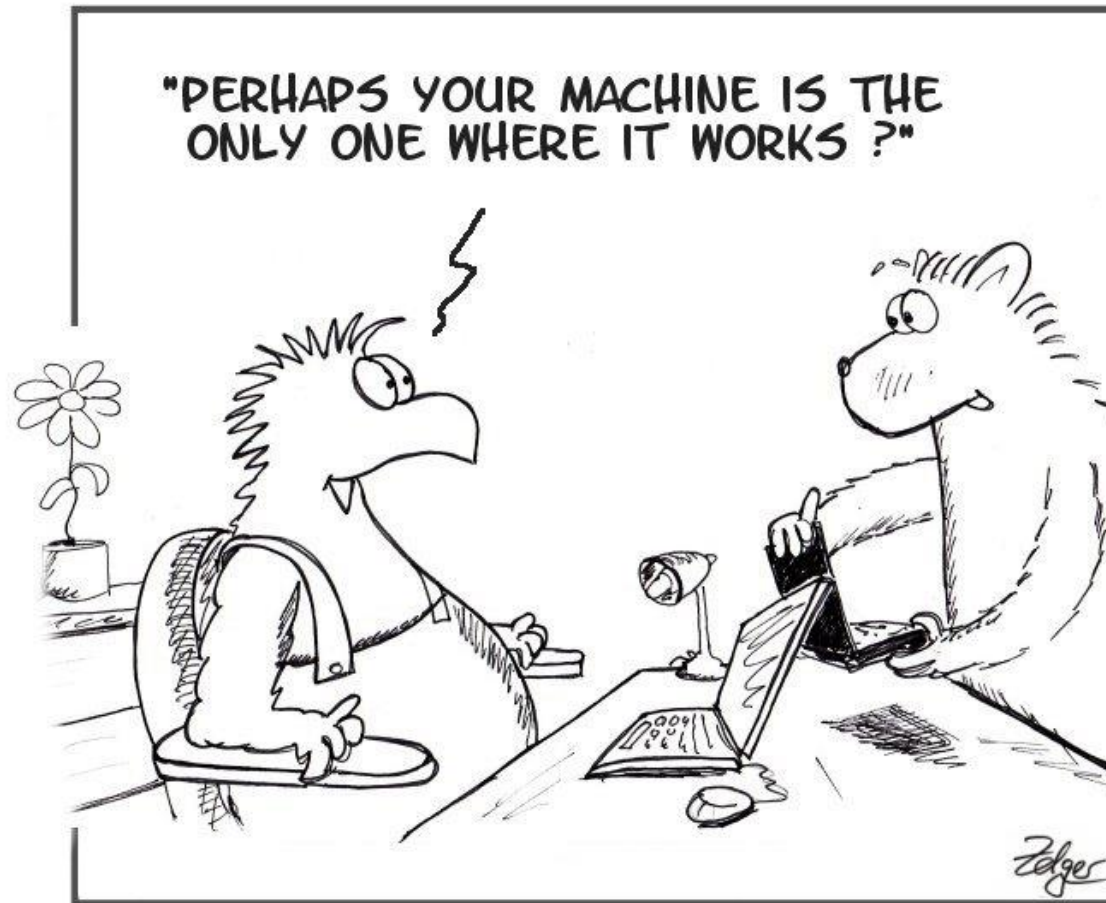
Driftsättning



+



C.A.G



It works on my machine

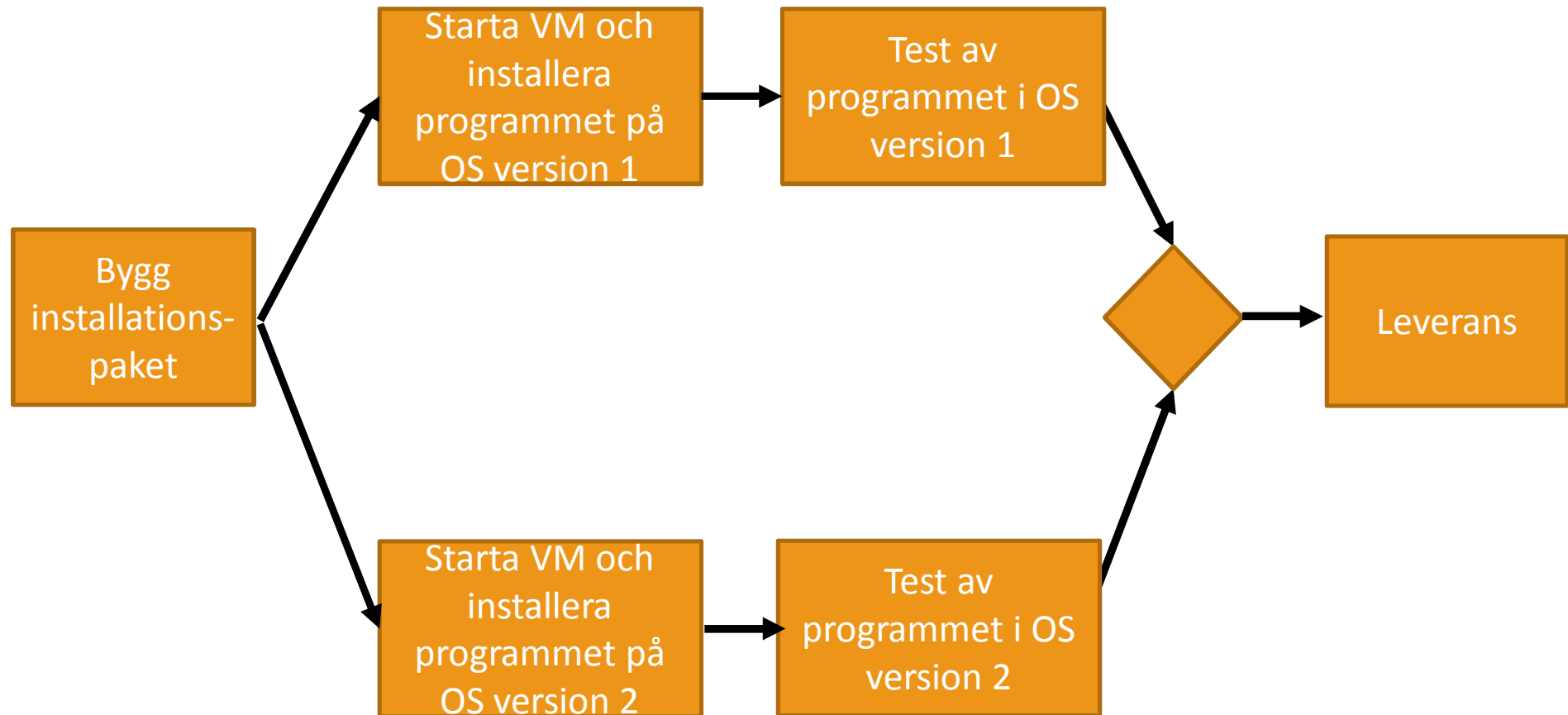
C.A.G



+



# Ex. Test av installation



C.A.G



[fredrik.edling@cag.se](mailto:fredrik.edling@cag.se)

072-236 68 29

C.A.G