

## Curs 3

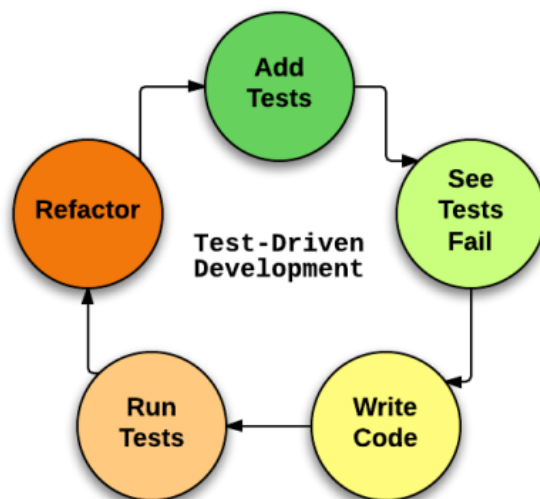
### Programare modulară

### Dezvoltare dirijată de teste (TDD)

— teste automate (chiar înainte de implementare)

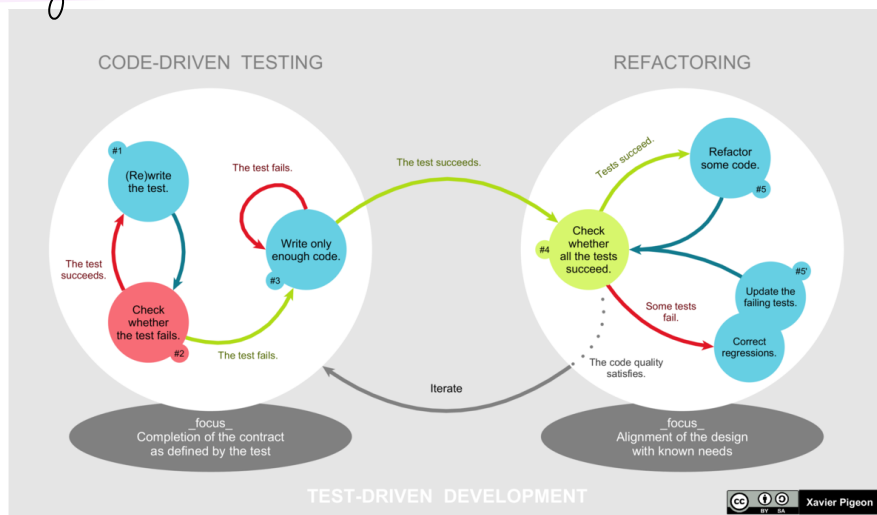
Pașii TDD pentru crearea unei funcții:

- Adaugă un test – creează teste automate
- Rulăm toate testele și verificăm ca noul test pică
- Scriem corpul funcției
- Rulăm toate testele și ne asigurăm că trec
- Refactorizăm codul



- Pas 1: facem testul prima dată  
 Pas 2: rulăm testele → pică testele  
 Pas 3: scriem funcția  
 Pas 4: rulăm testele → trec testele  
 Pas 5: REFACTORIZĂM CODUL

### Refactorizare



= restructurarea codului  
 alterând structura internă fără  
 a modifica comportamentul  
 observabil

Scop: codul va fi mai ușor de:

- înțeles
- întreținut
- extins

Semne: (Code Smell)

- cod duplicat
- metode lungi
- liste lungi de parametri
- instrucțiuni condiționale care determină diferite de comportament

- ex):
- redenumim funcția/variabila cu nume sugestiv
  - extragerea de metode (parte din funcție se transformă într-o funcție separată / expresie → funcție)
  - substituie algoritmi

## Funcții pure, funcții ca și obiecte

**Funcție pură** (pure function): funcție care nu are efecte secundare. Face o mapare între datele de intrare (parametrii funcției) și valoarea returnată. Nu modifică parametrii de intrare, nu folosește variabile globale, nu are efecte secundare. Pentru același intrări produce tot timpul același rezultat.

pt că: → ușor de înțeles / folosit / refolosit  
 → ușor de testat  
 → stau la baza programării funcționale  
 → facilitează paralelizare / memoizare

! divide et impera → sparge-ți problema în subprobleme și funcția în subfuncții

! Funcțiile în Python sunt obiecte (first class objects)

- se pot asigna la variabile
- se pot folosi ca parametri la alte funcții
- se pot adăuga în container

## Programare modulară

Descompunerea programului în **module** (componente separate interschimbabile) având în vedere:

- separarea conceptelor
- coeziunea elementelor dintr-un modul
- cuplarea între module
- întreținerea și reutilizarea codului
- arhitectura stratificată

Modulul este o unitate structurală separată, interschimbabilă cu posibilitatea de a comunica cu alte module. O colecție de funcții și variabile care implementează o funcționalitate bine definită.