



Programozási technológia

TDD

Dr. Szendrei Rudolf
ELTE Informatikai Kar
2018.

Teszt vezérelt szoftverfejlesztés

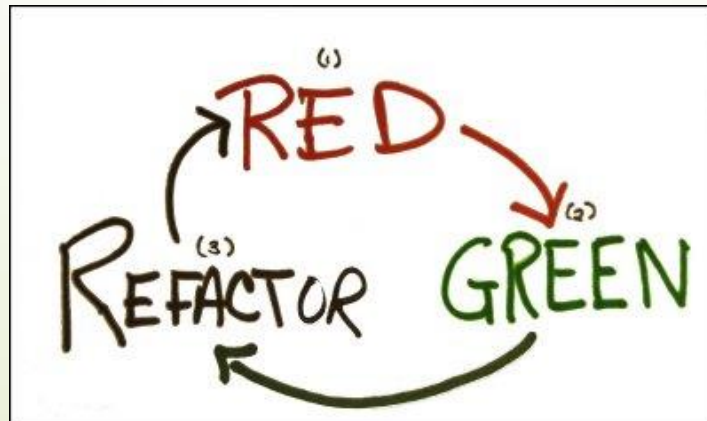
Definíció

- A TDD a szoftverfejlesztés egy evolúció alapú megközelítése, amely ötvöződik az „előbb-teszteljünk” fejlesztési módszertannal. A tesztek már azelőtt megírjuk, mielőtt elkészítenénk azt az éppen elegendő kódot, ami átmegy a teszten, majd ezt refaktoráljuk.
- Ez is egy módja annak, hogy átgondoljuk a követelményeket vagy a tervet, mielőtt megírnánk a működő kódot.

Uncle Bob törvénye (Robert C. Martin)

A TDD három alapszabálya

- NE kódoljunk semmit, kivéve ami ahhoz kell, hogy a programunk átmenjen a sikertelen teszten.
- Tesztből csak éppen elegendő mértékűt írjunk a hiba demonstrálásához.
- Csak annyi kódot írjunk, amennyi éppen elegendő a sikeres teszthez.



+1 szabály

- Legyünk óvatósak az előbbi szabályokkal!
- Azt mondják általában, hogy csak annyit kódoljunk, ami pillanatnyilag épp szükséges. Ez nem elég!
- Végeredményként egy tökéletesen struktúrált és refaktorált kódot kell kapnunk, ami nem csak a teszteket elégíti ki.
- **A tesztek a lehetséges használati eseteknek csak egy részhalmazát fedik le.**
- A végső kódnak jól kell működnie az összes lehetséges tesztesetben. Ne felejtjük ezt el!

Bowling példa

- Készítsünk egy programot, ami kiszámítja a Bowling játék során szerzett pontunkat.
- Tíz mezőt kell teljesítenünk, és minden mezőben két gurítási lehetőségünk van.
- Ha tarolunk (strike), a következő két gurítás eredménye hozzáadódik a tarolás eredményéhez.
- Ha másodikra döntjük le a bábukat (spare), akkor csak a következő gurítás eredménye adódik pluszban a pontszámokhoz.
- A 10. mező hibátlan teljesítésénél bónusz dobást kapunk (strike-nál kettőt, spare-nél egyet).

1	4	4	5	6	▲	5	▲	0	1	7	▲	6	▲	2	▲	6
5	14	29	49	60	61	77	97	117	133							