

# Testing

## 1. Unit Tests (Modultests)

- Unit tests dienen zum Testen verschiedener Abläufe eines Programms
- Die beste Art dieses Testens ist:  
Der Black-Box-Test:
  - Es wird eine Umgebung erzeugt, in der die zu testenden Funktionen mit Parametern aufgerufen werden und prüft die (eventuellen) Rückgabewerte der Funktionen und die Umgebung auf Veränderungen
  - Am besten in der gleichen Sprache wie das Programm geschrieben
  - Es wird empfohlen äußerst brutale, sogenannte Negativ-Tests zu benutzen, in denen das Programm mit den extremsten Fällen wie falschen Parametern/... getestet wird. Es ist zu empfehlen dies von anderen Programmierern als den machen selbst schreiben zu lassen
  - Beispiel:  
Eine Variable X soll um 2 erhöht werden. Das Testprogramm legt also fest, welchen Wert X bekommt, z. B. 5, ruft dann die Funktion auf und prüft danach wieder, ob X nun 7 ist oder nicht und gibt dann aus ob es einen Fehler gab.

## 2. Code Coverage (Testabdeckung)

- Code Coverage ist das Testen gewisser Codeabschnitte auf Stabilität
- Wichtig hierfür ist „code coverage analysis“:
  - o Gebiete des Programmcodes finden, die noch nicht ausgiebig getestet wurden
  - o Testfälle erfinden um die Abdeckung des zu testenden Codes zu erhöhenUmso höher der Grad des abgedeckten Programmcodes ist, desto besser
- Programme für Java:
  1. Cobertura (kostenlos und in Eclipse verfügbar durch update)
  2. CodeCover (kostenlos und in Eclipse verwendbar)
  3. Clover (funktionstüchtig in Eclipse aber Kostenpflichtig {30 Tage Trial})