

# **Mutaatiotestaus oliojärjestelmissä**

Eveliina Pakarinen

Referaatti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Helsinki, 9. syyskuuta 2015

Olioperustaisen ohjelmoinnin kehityksen myötä klassisia ohjelmistojen testausmenetelmiä on jouduttu sopeuttamaan uusiin vaatimuksiin, joita oliojärjestelmien kattava ja laadukas testaaminen vaatii. Vaikka olioperustainen ohjelmoi ratkaisee joitakin proseduraalisen ohjelmoinnin suunnittelu- ja toteutusongelmia, se tuo mukanaan myös uusia haasteita, jotka vaativat uusien testaus- ja analysointimenetelmien kehittämistä [MP08].

Testausta on käytetty ohjelmistokehityksessä ohjelmistokoodin laadun varmistamiseen ja samalla helpottamaan virheiden havaitsemista jo kehitysvaiheen aikana. Testien avulla on pyritty varmistamaan, että ohjelma toimii halutulla tavalla.

PALJON TEKSTIÄ KERTOEN black-white-box testauksista ja yksikkö/jne testauksen tasoista. Yleisesti testaamisesta.

Ohjelmiston laatua parannettaessa seuraava vaihe on varmistaa (MUOKKAA TÄTÄ JUTTUA JOTENKIN JÄRKEVÄMMÄKSI), että ohjelmistoa varten tehdyt testit ovat laadukkaita ja että ne havaitsevat kattavasti ohjelmistossa mahdollisesti esiintyvät virheet ja ongelmat. Yksi mahdollinen tapa testikoodin laadun arvioimiseksi on *mutaatiotestaus* (*mutation testing*), jonka tuottaman *mutaatiopistemäärän* (*mutation adequacy score*) avulla voidaan mitata, kuinka hyvin ohjelmiston testeillä voidaan havaita ohjelmakoodissa olevia vikoja. Mutaatiopistemäärä kertoo siis testauksen kohteena olevien testien laadukkuudesta. Mutaatiotestaus on virheperustainen (fault-based? mikä olisi parempi suomennos) testausmenetelmä, jonka taustaperiaatteena on ohjelmoijien tekemien ohjelmointivirheiden simulointi [JH11].

## Lähteet

- JH11 Jia, Yue ja Harman, Mark: *An Analysis and Survey of the Development of Mutation Testing*. IEEE Trans. Softw. Eng., 37(5):649–678, syyskuu 2011, ISSN 0098-5589. <http://dx.doi.org/10.1109/TSE.2010.62>.
- MP08 Mariani, Leonardo ja Pezze, Mauro: *Testing Object-Oriented Software*, luku Emerging Methods, Technologies and Process Management in Software Engineering. Wiley-IEEE Computer Society Press, 2008.