

58160 Ohjelmoinnin harjoitustyö

Testaus
30.3.2009

Tuntiop. Sami Nikander
sami.nikander@helsinki.fi

Testaus

- Ohjelman systemaattista tutkimista aiemmin *tuntemattoman* virheen löytämiseksi
- **Asenne: Yritetään saada ohjelma olemaan toimimatta niin kuin pitäisi**
 - Nurinkurista: "Onnistunut" testaus on sellainen, joka paljastaa virheen!
 - Testaus joka ei löydä virhettä, on epäonnistunut = ei todista virheettömyyttä!
 - Käytännössä pitää lopettaa sovittuun rajaan, eli viimeinen testauskierros on aina jossain määrin "epäonnistunut"!

Testaus

- **Ks. FT Juha Tainan kalvot testauksen perusteista:**

http://www.cs.helsinki.fi/u/nikander/oht/testauksen_perusteet.pdf

- **Edelliseen liittyvä harjoitelma:**

http://www.cs.helsinki.fi/u/nikander/oht/testaus_harjoitus.pdf

Harjoitustyön testaus

- Testattava *ainakin keskeiset osat* ohjelmasta
- Ohjaaja määrittelee opiskelijan kanssa, mitkä osat pitää testata (riippuu työstä)
- Esim. haaraumakattavuus on hyvä strategia
 - Testitapaukset lähdekoodin pohjalta (= white box)
 - Jokainen haarauma tulee olla läpikäyty vähintään kerran testauksen aikana (= haaraumakattavuus)
 - Ei testata esim. ohjelman käyttöliittymän painikkeita, vaan ohjelmakoodin metodeita
 - Tutkittava metodien lohkorakennetta sekä ehto-/toistolauseita: jokaisessa lohossa on käytävä

Harjoitustyön testaus

- **Lopullisen työn testaus on ERI ASIA kuin koodatessa tapahtuva intuitiivinen (ad hoc) testaaminen**
- **Testauksen on oltava systemaattista ja kattavaa: sinun on pystyttävä perustelemaan miksi teit mitäkin**
- **Testauksen on oltava toistettavissa: ohjaajan on pystyttävä ajamaan samat testit ja saatava samat tulokset**
- **Testausta varten laadit testauskoodia, jossa metodeille syötetään parametrit, joilla jokainen lause tulee suoritettua**
- **Kun testaus on suoritettu, koodia ei saa muuttaa - muuten testitulokset eivät enää päde**

Testauskoodi

- **Testilauseet main-metodissa**
 - Normaaliajossa vain pääluokan main suoritetaan, joten muihin luokkiin voi tehdä "ylimääräisen" main-metodin
 - Esimerkki: <http://www.cs.helsinki.fi/u/jnenonen/alabra/SuperTankkeri.java>
- **Junit-testiajurit myös hyvä työkalu (ei pakollinen)**
 - Kotisivu: <http://www.junit.org/>
 - Tutoriaali: <http://open.ncsu.edu/se/tutorials/junit/>
- **Javan assert-lausetta voi käyttää testauskoodissa**
 - <http://java.sun.com/javase/6/docs/technotes/guides/language/assert.html>

Testiraportti

- Testauksen periaatteet, testitapaukset ja testiajojen tulokset dokumentoidaan testiraportiksi
- Testausperiaatteet:
 - Mitä strategiaa/strategioita on käytetty
 - Miten kattava testaus mielestäsi on jne.
- Jokaisesta testitapauksesta dokumentoidaan:
 - Testattava toiminto (tai: "testin otsikko")
 - Testin tarkoitus: mitä testillä yritetään saada selville?
 - Testin alkutilanne: mitä on tapahtunut ennen testiä?
 - Testin kulku: mitä testissä tehdään?
 - Odotettu tulos: mitä testissä pitäisi tapahtua?
- JUnit-testikehystä käytettäessä osan em. kohdista voi dokumentoida hyvin kommentoidulla JUnit-lähdekoodilla

Testiraportti

- **Aidon testiajon data liitteeksi:**
 - Testin suoritus ja kulku, syötteet ja tulosteet ym.
 - JUnit-testien osalta riittää toimittaa testiajurien lähdekoodi, jonka ohjaaja suorittaa
- **Odotetun tuloksen oltava sama kuin saatu tulos. Virheet korjataan. Viimeinen testikierros on siis epäonnistunut (!) testaus, jossa ei enää löydy virheitä.**
- **Mikäli et ehdi tai osaa korjata virhettä, kysy hyväksyykö ohjaaja korjaamisen sijaan myös pelkän dokumentoinnin**
 - Tällöin ohjelmaan jää siis (dokumentoitu) bugi
 - Dokumentoitu bugi on arvostelun kannalta parempi kuin dokumentoimaton, jonka ohjaaja kuitenkin löytää tarkastuksessa...
 - Bugista kerrottava käyttöohjeessa, koodissa ja tekn. dokumentissa

