

Arvioinnin kehittäminen ViSCoS-opinnoissa

Osku Kannusmäki

28. toukokuuta 2008

Joensuun yliopisto
Tietojenkäsittelytiede
Pro gradu -tutkielma

Tiivistelmä

ViSCoS-opinnoiksi kutsutaan 25 ECTS-opintopisteen laajuisia Internetin välityksellä opetettavia tietojenkäsittelytieteen perusopintoja. Opinnoissa kurssien arviointi suoritetaan pääosin kirjallisten tenttien perusteella. Tenttien postittaminen ja tenttitilaisuuksien järjestäminen aiheuttaa aikatauluongelmia tenttien arvostelussa. Perinteisiin tentteihin liittyy myös muita ongelmia ja etenkin ohjelmointitaidon arviointiin perinteinen tentti soveltuu kankeasti. Opintojen arviointiin käytettävien menetelmien valinta vaikuttaa pitkälti siihen, mitä opiskelijat muistavat opinnoista. ViSCoS-opintojen arviointia onkin syytä kehittää. Arviointi muuttuu luontevammaksi ja joustavammaksi, kun kurssien harjoitustehtävät otetaan suuremmaksi osaksi arviointia ja arviointi muutetaan kokonaan verkkopohjaiseksi. Tässä työssä esittelen tarkemmat perusteet arviointimenetelmien valinnoille ja uudistetut arviointimenetelmät ViSCoS-opinnoille.

ACM-luokat (ACM Computing Classification System, 1998 version): K.3.1 **Computers and Education**: Computer and information Science Education – computer science education

Avainsanat: arviointi, arvostelu, verkko-opinnot, ViSCoS

Esipuhe

Viimeinen puristus opintojen loppuun saattamiseksi on nyt vihdoinkin takana pro gradun myötä. Tutkielman kirjoittaminen oli pitkälinen prosessi ja sen ajalle elämä oli varannut monta ylä- ja alamäkeä, jotka vaikuttivat motivaatiooni ja työskentelyyni. Nyt on kuitenkin aika kääntää katse kohti uusia haasteita ja tulevaisuutta.

Tahdon kiittää ohjaajaani Jussi Nuutista arvokkaasta tuesta, kärsivällisyydestä ja hyvästä ohjauksesta työskentelyni aikana. Kiitos kuuluu myös Ilkka Jormanaiselle, joka antoi työn loppuvaiheessa asiantuntevaa palautetta. Oman kiitoksensa ovat ehdottomasti ansainneet myös perheeni ja ystäväni, joiden ansiosta sain välillä pidettyä ansaittua taukoa. Sain heiltä myös tukea kun oikeat sanat tuntuivat olevan kiven alla. Suurin kiitos kuuluu kuitenkin Selmalle, joka jaksoi kehrätä ja torkkua vieressäni sohvalla kirjoittaessani ja välillä herättää minut ajatuksistani vaatimalla silitämistä. Kiitos teille kaikille!

Järvenpäässä 28.5.2008

Osku Kannusmäki

Sisältö

1 Johdanto	1
1.1 Tutkimuskysymykset	3
1.2 Tutkielman rakenne	4
2 Opiskelijan osaamisen arviointi	5
2.1 Perinteinen tentti	6
2.2 Kaksoisrakenne	8
2.3 Vilppi	11
2.4 Arvioinnin suhde opetussuunnitelmaan	12
2.5 Opiskelijan arviointiin käytettävät menetelmät	13
2.6 Hyvän arviointimenetelmän ominaisuudet	16
3 Tietojenkäsittelytieteen verkko-opintojen arviointi	18
3.1 Verkko-oppimisympäristö arvioinnissa	19
3.2 Vilppi tietojenkäsittelytieteen verkko-opinnoissa	21
3.3 Ohjelmoinnin osaamisen arvioinnin erityishaasteet	24
3.4 Tietokoneavusteinen arviointi	27
3.5 Verkkotentit ja tunnistautuminen verkossa	33
4 Yleisesti käytettyjen arviointimenetelmien esittely	36
4.1 Perinteinen tentti ja sen johdannaiset	36
4.2 Etätentit	38
4.3 Muut arviointimenetelmät	39
4.4 Arvioinnin hajauttaminen	43
5 ViSCoS-opinnot	44
5.1 ViSCoS-opintojen historia	44
5.2 ViSCoS-kurssit	46
5.3 Opiskelijan osaamisen arviointi nykyisissä ViSCoS-opinnoissa	49
6 Arvioinnin kehittäminen ViSCoS-opinnoissa	51
6.1 Hyvän ViSCoS arviointimenetelmän ominaisuudet	51
6.2 ViSCoS-kursseilla nykyisin käytettävät arviointimenetelmät	54
6.3 Perinteisistä tenteistä verkkoarviointiin	60
6.4 Suositellut arviointimenetelmät ViSCoS-kursseille	62
6.5 Uudet arviointimenetelmät ja palaute ViSCoS-opinnoissa	70
7 Yhteenveto	73
Liite 1: Esimerkki matriisiarvioinnista	84

1 Johdanto

ViSCoS eli Virtual Studies of Computer Science, sisältää Joensuun yliopiston Tietojenkäsittelytieteen ja tilastotieteen laitoksen 25 ECTS-opintopisteen laajuiset tietojenkäsittelytieteen perusopinnot¹. Opinnot tarjotaan opiskelijoille lähes kokonaan etäopetuksena Internetin välityksellä. Vuonna 2006 opintoja alettiin järjestää myös englanniksi.

Opiskelijoiden fyysinen läsnäolo on ollut ViSCoS-opinnoissa välttämätöntä ainoastaan tenttitilaisuuksissa, joihin opiskelijat on koottu suorittamaan kirjallisen tentin kurssilla käsitellyistä asioista. Osa kursseista arvioidaan muilla menetelmillä, mutta selvästi eniten kursseja arvostellaan tällä hetkellä tenttien perusteella. Tentillä mitataan sitä, kuinka hyvin opiskelija on ymmärtänyt ja omaksunut kursseilla opetettavan asian. Opiskelijat ja ViSCoS-opintojen suoritumahdollisuutta tarjoavat yhteistyöoppilaitokset ovat kokeneet tenttitilaisuuksissa läsnäolon ja tilaisuuksien järjestämisen monimutkaiseksi.

Tenttien laatiminen on ongelmallista, sillä eri tenttitilaisuuksien järjestäjille sopivaa tenttiaikaa on vaikea sovittaa edes Suomen sisällä, saati useammassa maassa yhtä aikaa. Tentti tulisi järjestää joka paikassa samaan aikaan, jotta tentin myöhemmin tekevät eivät voisi tiedustella tenttikysymyksiä etukäteen jo tenttiin osallistuneilta. Useampien tenttien ja kysymysten laatiminen on hyvin hankalaa ja joskus ViSCoS-opettajat ovat joutuneet laatimaan samalle päivälle jopa neljä erilaista tenttiä samasta kurssista aikatauluongelmien takia. Tämä on turhaa opettajien resurssien haaskausta. Oman ongelmansa aiheuttaa ulkomailla tenttiminen. Eri aikavyöhykkeillä tenttien samaan aikaan järjestäminen on lähes mahdotonta.

ViSCoS-opintojen parissa työskentelevien opettajien työaika kuluu pääasiassa opiskelijoiden viikoittaisten tehtävien tarkastamiseen sekä tenttien laatimiseen ja tarkastamiseen. Varsinkin kursseilla, joille osallistuu yli sata opiskelijaa, tenttien tarkastaminen vie kurssia kohti käytettävästä opettajien työajasta hyvin suuren osan. Tenttien tarkastusaika on tarkasti rajattu. Kun tenttejä on tarkastettavana samaan aikaan useita, ViSCoS-opettajan koko työpanos menee tenttien tarkastamiseen. Tällöin samaan aikaan käynnissä olevat kurssit jäävät opettajan ajan puutteen vuoksi vähemmälle huomiolle.

¹Aikaisemmin opintoja nimitettiin Virtuaaliapprobatur-opinnoiksi. Opintojen laskennallinen laajuus oli 15 opintoviikkoa.

Joensuun yliopiston opintosuorituksia ja opiskelijan oikeusturvaa koskeva johtosääntö velvoittaa opettajan julkistamaan kirjallisen ja suullisen kuulustelun tuloksen viimeistään kahden viikon kuluttua kuulustelusta (Joensuun yliopisto, 2006). Esseemuotoisten suoritusten tulokset tulee ilmoittaa viimeistään kuukauden kuluttua kuulustelusta. Myös kirjallisten kuulustelujen tarkastusaikaa on mahdollista pidentää enintään kuukauden mittaiseksi tiedekunnan dekaanin tai laitoksen johtajan erityisluvalla.

Koska ViSCoS-kuulustelut ovat pääosin kirjallisia tenttejä, niiden tulokset pitää julkaista viimeistään kahden viikon kuluttua suorituksesta. Tähän on mahdollista hakea poikkeusta, mutta opintojen sujuvuuden vuoksi opiskelijoiden tulisi kuitenkin saada tieto opintosuorituksen tuloksesta mahdollisimman pian. Lisäksi etäopinnot pyritään järjestämään mahdollisimman samanlaisina kuin normaalit tietojenkäsittelytieteen kontaktiopinnot. Jos lähiopetukseen osallistuvien tentit tarkastettaisiin nopeammassa aikataulussa, lähiopetukseen osallistuvat saisivat etua ViSCoS-opintoihin osallistuviin opiskelijoihin nähden.

ViSCoS-opinnot suoritetaan pääosin muualla kuin Joensuussa ja osa opiskelijoista suorittaa opintonsa myös ulkomailla. Tästä johtuu, että tenttien postitusajat venyvät usein kohtuuttoman pitkiksi. Valitettavasti opiskelijoiden tenttivastauksia ei voida lähettää sähköisenä, sillä esimerkiksi erilaisilla kuvanlukijoilla digitoidut tentit asettavat opiskelijat eriarvoiseen asemaan. Huippuskannerilla ja huolellisella skannaamisella on kylä mahdollista saada erittäin hyviä digitaalisia kopioita tenttivastauksista, mutta ongelmaksi muodostuukin heikommilla kuvanlukijoilla skannatut tentit, joissa olevasta opiskelijan tekstistä opettaja ei välttämättä saa tarpeeksi hyvin selvää.

Koska ViSCoS-kurssien arvostelua tarkastellaan tämän työn puitteissa, myös koko tentin käyttäminen opiskelijan osaamisen arviointikeinona on tässä vaiheessa perusteltu asettaa kyseenalaiseksi. Erityisen ongelmallisena ViSCoS-opettajat ovat kokeneet ohjelmoinnin osaamisen arvioinnin tenttien avulla. Perinteisiä tenttejä kohtaan on esitetty myös melko paljon kritiikkiä ja useita muitakin opiskelijan osaamista mittaavia menetelmiä on kehitetty.

Käsittelen² tässä työssä erilaisia vaihtoehtoja ViSCoS-opintojen tenttien korvaamiseksi mahdollisuuksien mukaan joustavammalla sekä soveltuvammalla menetelmällä, kuin lähiopetuksena toteutettavat tenttitilaisuudet. Tarkastelen erilaisia *osaamisen arvioin-*

²Olen toiminut opettajana ViSCoS-opinnoissa vuodesta 2002 vuoteen 2006 saakka, joten ViSCoS-opintojen käytännön ongelmat ovat minulle tuttuja.

timenetelmiä, eli erilaisia tekniikoita joilla ViSCoS-opettajat voivat seurata opiskelijan menestymistä opinnoissa ja opettajien asioiden hallitsemista sekä ymmärtämistä.

1.1 Tutkimuskysymykset

Edellä esitettyjen kehitystarpeiden vuoksi tässä tutkielmassa pyrin löytämään vastaukset seuraaviin tutkimuskysymyksiin:

1. Millaisia yliopistotason tietojenkäsittelytieteen verkko-opetukseen soveltuvia opiskelijan osaamista mittaavia menetelmiä on olemassa?
2. Millaisia ominaisuuksia vaaditaan ViSCoS-opintojen aikana käytettäviltä arviointimenetelmiltä?
3. Mitkä tässä työssä käsitellyt menetelmät soveltuvat ViSCoS-opintojen ja -kurssien arvioinnissa käytettäväksi ja miten niitä voitaisiin hyödyntää?

Opiskelijan osaamisen arviointiin on olemassa lukuisia menetelmiä. Vastatakseni ensimmäiseen tutkimuskysymykseen käyn läpi kirjallisuudessa esiteltyjä arviointimenetelmiä, joita voidaan hyödyntää tietojenkäsittelytieteen verkko-opetuksessa.

Käytettävä arviointimenetelmä vaikuttaa merkittävästi opiskelijoiden oppimiseen (Gibbs ja Simpson, 2004). Toisaalta sekä ViSCoS-opintojen verkko-opetusluonne että opintojen sisältö asettavat vaatimuksia soveltuvien arviointimenetelmien valinnalle. Vastatakseni toiseen tutkimuskysymykseen selvitän, millaisia vaatimuksia ViSCoS-opinnoissa käytettävän arviointimenetelmän tulee täyttää.

Esittelen ja analysoin kahden ensimmäisen tutkimuskysymyksen valossa erilaisia arviointimenetelmiä, sekä sitä millaiset menetelmät soveltuvat käytettäväksi ViSCoS-opinnoissa. Viimeisellä tutkimuskysymyksellä pyrin löytämään konkreettisen tavan kehittää ViSCoS-opintojen arviointia. Vastauksena tutkimuskysymykseen käyn läpi millaisia menetelmiä ViSCoS-opintojen arvioinnissa tulisi tulevaisuudessa käyttää ja mitä etuja ne tarjoavat suhteessa nykyiseen arviointikäytäntöön.

1.2 Tutkielman rakenne

Luvussa 2 käsittelen yleisesti osaamisen arviointia sekä siihen liittyviä asioita. Käyn läpi sen, mitä arvioinnilla tarkoitetaan ja mitä arviointi merkitsee opiskelijan ja opettajan kannalta. Keskityn luvussa myös tarkemmin perinteiseen tenttiin, sekä siihen liittyviin ongelmiin. Esittelen luvussa myös kaksoisrakenne-termin ja miten se vaikuttaa arviointiin. Paneudun myös vilpin ongelmaan sekä siihen, miten opetussuunnitelma vaikuttaa arviointiin. Tarkastelen myös arviointimenetelmien jaottelua sekä hyvän yliopistollisen arviointimenetelmän ominaisuuksia.

Avaan arvioinnin järjestämistä erityisesti tietojenkäsittelytieteen verkko-opintojen näkökulmasta luvussa 3. Käyn läpi verkko-oppimisympäristön merkitystä arvioinnin toteuttamiseen, vilpin esiintymistä tietojenkäsittelytieteen opetuksessa sekä keinoja vilpin vähentämiseen. Keskityn luvussa myös ohjelmoinnin oppimiseen ja arviointiin. Tarkastelen näiden asioiden lisäksi tietokoneavusteista arviointia tietojenkäsittelytieteen verkko-opinnoissa sekä verkkotenttimiseen liittyviä ongelmia.

Luvussa 4 käsittelen tarkemmin yksittäisiä arviointimenetelmiä, sekä niiden hyödyntämistä tietojenkäsittelytieteen opetuksessa. ViSCoS-opinnot ja kurssit esittelen tarkemmin luvussa 5. Käyn luvussa yleisesti läpi myös nykyisen ViSCoS-opintojen arvioinnin.

Arvioinnin kehittämiseen ViSCoS-opinnoissa keskityn tarkemmin luvussa 6. Tarkastelen luvussa millaisia ominaisuuksia hyvältä ViSCoS-arvioinnilta edellytetään ja miten nämä vaatimukset toteutuvat nykyisten arviointimenetelmien osalta. Kehitän luvussa uudistetut arviointimenetelmät ViSCoS-kurssien arviointiin ja käyn läpi sitä, millaisia etuja uudistetulla arvioinnilla on. Luvussa 7 esitän yhteenvedon tässä työssä käsitellyistä asioista sekä johtopäätöksistä.

2 Opiskelijan osaamisen arviointi

Opiskelijan *arvioinniksi*³ nimitetään prosessia, jossa mitataan sitä, miten hyvin opiskelija on omaksunut opetuksessa käsitellyn asian (Committee on the Foundations of Assessment, National Research Council, 2001). Arvioinnista on käytetty suomenkielisessä kirjallisuudessa myös termiä *evaluaatio* (Karjalainen, 2001).

Karjalainen (2001) erottaa selvästi toisistaan termit arviointi ja arvostelu. Arvostelulla viitataan kirjoittajan mukaan vahvasti arvosanojen antamiseen, kun taas termi arviointi voidaan käsittää laajemmin. Arvostelu eli opintosuorituksen arvioiminen arvosanalla voidaan toki laskea myös osaksi arviointia. Arvioijana voi toimia esimerkiksi opettaja, ryhmä, toinen opiskelija tai opiskelija itse. Opiskelijoiden arvioinnilla on pitkä historia. Nykyisenkaltainen arviointijärjestelmä on saanut alkunsa jo keskiajalla (Wilbrink, 1997).

Opiskelijan osaamista on arvioitu ja mitattu hyvin erilaisin keinoin. Opiskelijan osaamisen arviointi syntynyt opettajien tarpeesta tarkastella kuinka hyvin opiskelijat oppivat ulkoa opetettavia asioita (Wilbrink, 1997). Karjalainen (2001) kuitenkin kyseenalaistaa ulkoa osaamisen tarpeen nykyisissä yliopisto-opinnoissa. Kirjoittajan mukaan opintojen aika omaksuttavaksi tarkoitettu tietomäärä on niin valtava, että sen ulkoa oppiminen on mahdotonta. Uusia arviointijärjestelmiä syntyy jatkuvasti lisää. Arviointitekniikoissakin on huomattavissa erilaisia muotivirtauksia varsinkin sen suhteen, mikä tyyppistä oppimisteoriaa milloinkin painotetaan.

Opiskelijan arvioinnista on kehittynyt oma tutkimusala, jossa kognitiivisten tieteiden tutkimustuloksia pyritään hyödyntämään niin, että saadaan mahdollisimman kattava kuva siitä, miten hyvin opiskelija ymmärtää opetetun asian. Lisäksi erilaiset oppimisteoriat ja verkko-opetuksen muoti-ilmiöt vaikuttavat siihen minkälaisia oppimisen arviointikeinoja milloinkin käytetään.

Arviointimenetelmät riippuvat opetettavasta asiasta, sekä siitä, millaisia tietoja opiskelijan oppimisesta halutaan nostaa esille (Committee on the Foundations of Assessment, National Research Council, 2001). Yleensä arvioinnista saatu tieto välitetään myös opiskelijalle joko sanallisena palautteena tai arvosanoin. Joskus tieto jää ainoastaan opettajan käyttöön hänen suunnitellessaan sitä, miten opetuksessa edetään.

³Englanniksi *assessment* tai *evaluation*, joista edellistä käytetään amerikanenglannissa ja jälkimmäistä brittiläisessä ja mannermaisessa kirjallisuudessa (Karjalainen, 2001)

Gibbs ja Simpson (2004) erittelevät artikkelissaan sitä, kuinka arviointi voitaisiin suorittaa siten, että siitä olisi mahdollisimman paljon etua myös opiskelijan oppimiselle. Nykyisin arviointia ei nähdäkään enää hyödylliseksi ainoastaan opettajan ja opetussuunnitelman toteutumista seuraavien kannalta, vaan arvioinnin tuloksia pyritään käyttämään mahdollisimman paljon ohjattaessa opiskelijoita opiskelemaan niitä asioita, jotka opettajan mielestä ovat tärkeimpiä (Committee on the Foundations of Assessment, National Research Council, 2001).

Karjalainen ja Kempainen (1994) korostavat koko arviointitilaisuuden tärkeyttä opiskelijan oppimisen kannalta. Sen sijaan, että opiskelija keskittyisi tenttimään kurssin ja unohtamaan sen, kirjoittajat suosittelevat erilaisten opettajan ja opiskelijan dialogia painottavien arviointimenetelmien käyttämistä. Kirjoittajien mukaan perinteisen tentin käyttäminen opetuksessa voi olla opintojen kannalta jopa haitallista, sillä opiskelija ei keskity asioiden oppimiseen vaan ulkoa opetteluun.

Luvussa 2.1 käsittelen yliopisto-opintojen yleisintä arviointitapaa eli perinteistä tenttiä sekä siihen liittyviä ongelmia. Esittelen luvussa 2.2 kaksoisrakente-termin ja käyn läpi sitä, miten se liittyy opintojen arviointiin. Vilpin merkitykseen opinnoissa ja arvioinnissa paneudun luvussa 2.3.

Opinnot suunnitellaan yleensä noudattamaan yleisiä opintosuunnitelmia. Tarkastelen luvussa 2.4 arvioinnin suhdetta opetussuunnitelmaan. Opiskelijan arviointiin käytettävien menetelmien luokitteluun ja ominaisuuksiin keskityn tarkemmin luvussa 2.5. Viimeiseksi luvussa 2.6 listaan hyvän yliopistollisen arviointimenetelmän perusominaisuuksia.

2.1 Perinteinen tentti

Karjalainen (2001) on selvittänyt kyselytutkimuksessaan, että yleisin yliopisto-opinnoissa käytetty oppimisen arviointikeino on Suomessa koe tai tentti, jossa opiskelijalta kuulustellaan opetetun asian muistamista ja ymmärtämistä. Hän kuvaa väitöskirjassaan perinteistä tenttiä seuraavasti:

- *tentin ulkoiset puitteet rajoittuvat valvottuun työskentelyyn tenttialissa, -opiskelijalla on käytössään kirjoitusvälineet, oma pää ja tehtäväpaperi,*
- *työaika on tiukasti ennalta rajattu, ongelmatilanteesta selviytymiseen liittyvä viestintä ja kommunikointi (avun pyyntö, avun antaminen) on katkaistu sekä tenttijäkollegoihin että tentin valvojan/tentaattoriin,*
- *tenttisäännöstöön kuulumattomien työvälineiden käyttö on rangaistuksen uhalla kielletty,*
- *tenttijällä ei tenttitilanteessa ole mahdollisuutta vaikuttaa tilanteen kulkuun ja omaan selviytymiseensä muulta osin kuin annetun tehtävän ratkaisuun liittyen oman muistinsa, päättelykykynsä ja kokemuksensa nojalla.*

Karjalainen jatkaa:

Kyseessä on siis tavanomainen kynä-paperi-ulkomuisti -koe. Nimitän sitä perinteiseksi tentiksi, koska se on ollut Suomen yliopistoissa yleisimmin käytetty tentin toimintarakennetyyppi.

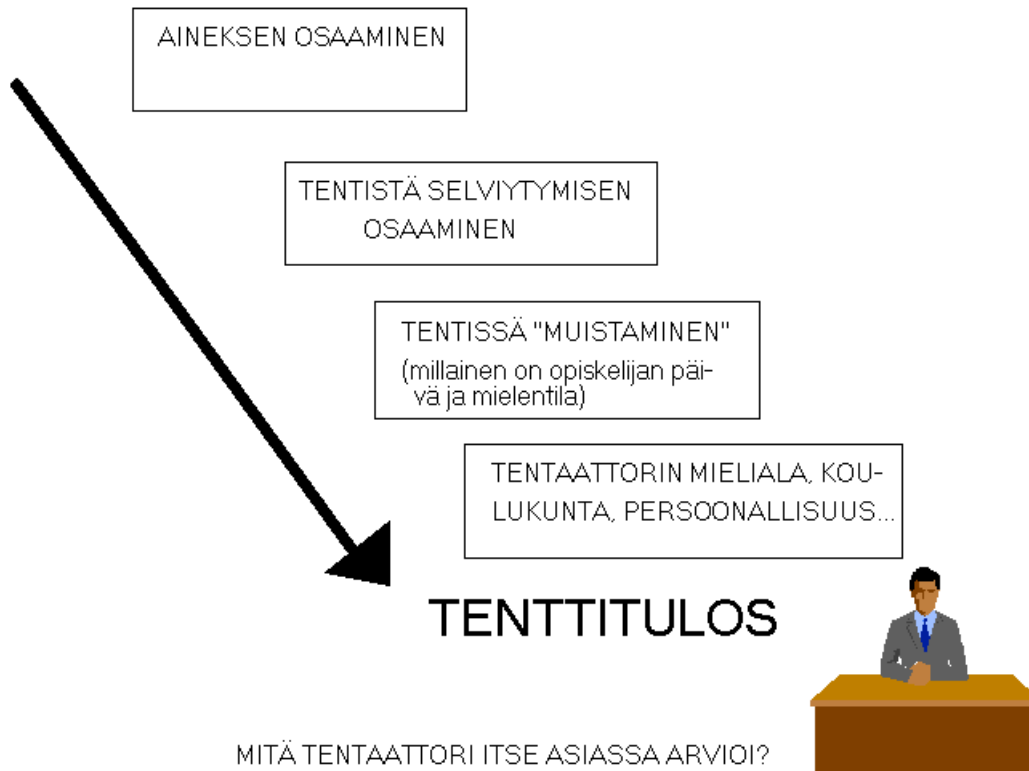
Karjalainen (2001) käyttää termiä *toimintarakenne*, johon hän määrittelee kuuluvaksi seuraavat piirteet: tentin ulkoiset puitteet (esimerkiksi luentosali ja tentin valvonta), säännöt joita opiskelijan tulee noudattaa tentissä (esimerkiksi työaika, työohjeet, juridiset säännöt), tenttijän ja tentaattorin vuorovaikutusmalli (esimerkiksi roolit, tehtävät ja viestintämahdollisuus), tenttijän työvälineet, tenttijän toiminnan vapausasteet (mahdollisuus itseohjautuvuuteen ja itsenäiseen päätöksentekoon ongelmatilanteessa edellä mainittujen tekijöiden, esimerkiksi työskentelytilan suhteen).

Karjalainen (2001) kiteyttää perinteiseen tenttiin liittyvät ongelmat seuraavaan kysymykseen:

Onko olemassa sellaista ammatillista tai tieteellistä ammattikäytäntöä, jonka toimintarakenne kokonaisuutenaan vastaisi perinteisen tentin toimintarakennetta tai olisi sille analoginen?

Perinteiseen tenttiin liittyvän arvioinnin ongelmat käyvät ilmi kuvasta 1. Perinteisen tentin merkitystä arviointikeinona ei pidä kuitenkaan sivuuttaa. Tentit ovat niin yleises-

sä käytössä, että ne täytyy ottaa mukaan tarkasteluun. Suurin osa yliopistossa järjestettävistä kursseista arvioidaan ja arvostellaan tentein. Myös ViSCoS-opinnoista suurin osa arvioidaan tällä hetkellä tenttien perusteella.

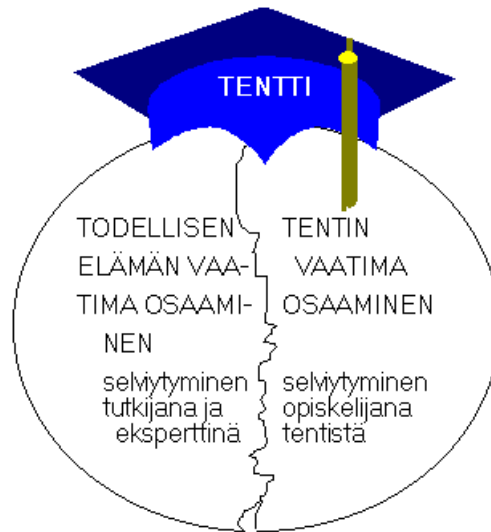


Kuva 1: Tentillä suoritettavan arvioinnin ongelma (Karjalainen ja Kempainen, 1994)

2.2 Kaksoisrakenne

Karjalainen (2001) esittelee väitöskirjassaan *kaksoisrakenne*-termin. Kuvassa 2 on esitetty kaksoisrakenteen periaatteellinen ongelma. Kirjoittaja määrittelee kaksoisrakenteen olevan tentin sisällön, toimintarakenteen ja tenttijän kompetenssin pedagoginen "käänteisilmiö". Karjalainen käyttää tentti-termiä laajassa merkityksessä. Tenti tulee tässä yhteydessä ymmärtää käsittämään kaikki arviointiin käytettävät menetelmät aina perinteisestä tentistä harjoitustyöhön.

Karjalainen (2001) luettelee kaksoisrakenteeseen kuuluvaksi seuraavat ominaisuudet:



Kuva 2: Tentin kaksoisrakenne (Karjalainen ja Kempainen, 1994)

1. Tentistä selviytyminen ei välttämättä vaadi opetetun aineksen hallintaa, vaan opiskelijalle riittää äärimmäisessä kaksoisrakenteisuudessa oppia oikeat keinot tentistä suoriutumiseen.
2. Tenti vaatii tekijältään sellaisten työtapojen tai keinojen hallintaa, jotka eivät vastaa reaalielämän toimintatapaa. Tentistä suoriutuakseen opiskelija joutuu opettelemaan tentistä selviämiseen vaadittavan työtavan.
3. Opiskelija erottaa ”tenttiminän ja työminän” toisistaan. Vaikka opiskelija koki-sikin osaavansa käsiteltävät asiat, on eri asia kuinka hän uskoo suoriutuvansa tentistä.
4. Tenttiin osallistuminen voidaan nähdä pakollisena ”ritualistisena” toimintana, jossa alistutaan aktoriteetille. Tentin läpäisy on itsessään arvokasta. Varsinainen asioiden oppiminen on sivuseikka.
5. Tentin oppimistavoitteen ja tenttimiseen liittyvien toimintamallien välinen ristiriita. Vaikka tentillä itse asiassa tavoitellaan osioiden oppimista, saattaa kaksois-rakenteellisuuden myötä painotus siirtyä oppimisesta tenttiin liittyvien toiminta-mallien painotukseen.

Sekä tentin toimintarakenne että sisältö vaikuttavat siihen, missä määrin kaksoisraken-teisuutta tentissä esiintyy (Karjalainen ja Kempainen, 1994). Taulukossa 1 on esitetty

kaksoisrakenteen muodostuminen tentin sisällön ja toimintarakenteen yhteisvaikutuksesta.

Taulukko 1: Kaksoisrakenteen muodostuminen (Karjalainen ja Kemppainen, 1994)

KAKSOIRAKENTEEN ULOTTUVUUDET	TOIMINTARAKENNE - miten tenttijä saa toimia tentissä, mitä "työvälineitä" käyttää	
	TUTKIJAN / EKSPERTIN -aineistojen käyttö - asiantuntijan ohjaus - yhteistyö kollegojen kanssa -keskustelu, raportointi, kritiikki	PERINTEISEN TENTIN - vain kynä, paperi, tehtävät ja muisti
S I		
ASIAN HALLINNAN KESKEISYYS S Ä L	1) HYVÄ AKATEEMINEN TENTTI. - opettaa oikeita työtapoja - asia ymmärretään ja osataan EI KAKSOISRAKENNETTA	2) - opettaa väärää työtapoja - asia ymmärretään ja osataan -> sosiaalisesti negatiivinen kaksoisrakenne = opitaan eristyvän asiantuntijuuden malli
SELVIYTYMISKEINOJEN HALLINNAN KESKEISYYS T Ö	3) -opettaa oikeita työtapoja mutta siten, että opiskelijat eivät niiden kautta sitoudu sisältöön vaan käyttävät niitä hyväkseen sosiaalisina selviytymiskeinoina -> sisällöllisesti negatiivinen kaksoisrakenne	4) KAKSOISRAKENTEINEN TENTTI - opettaa väärää työtapoja - ei opita asiaa vaan selviytymiskeinoja

Kaksoisrakenteen esiintyminen tentissä on haitallista sekä opiskelijan että opiskelun kannalta. Opiskelijalle ei ole edullista, että hän joutuu opettelemaan tenteissä menestyäkseen sellaisia työtapoja ja selviytymismenetelmiä, joilla ei ole merkitystä työelämässä. Mitä vähemmän kaksoisrakennetta arviointiin sisältyy sitä lähempänä arviointi on todellista työelämän tilannetta, joissa opiskelija voi oppimiaan taitoja hyödyntää. Opettajan ja kurssin kannalta taas on turhaa, että opiskelija kuluttaa osan opiskeluun käytettävissä olevista resursseistaan opetettavan sisällön kannalta epäoleellisen asian harjoitteluun.

Karjalainen (2001) ja Karjalainen ja Kemppainen (1994) suosittelevat arvioinnin muuttamista sellaiseksi, että tenttimiseen liittyvä kaksoisrakenne voidaan järkevästi murtaa. Suurin osa luvussa 2.1 kuvatuista perinteisen tentin ongelmista perustuu juuri siihen kuuluvaan voimakkaaseen kaksoisrakenteeseen. Perinteinen tentti on itsessään hyvin keinotekoinen, joten kaksoisrakenne esiintyy siinä hyvin voimakkaasti.

Kaksoisrakenteeseen liittyy myös *piilo-opetussuunnitelman* käsite. Tentin kaksoisrakenne voidaan nähdä osana koko yliopiston piilo-opetussuunnittelmaa (Karjalainen, 2001). Piilo-opetussuunnitelmalla tarkoitetaan koko koulutusjärjestelmän tasolla ta-

pahtuvaa oheisoppimista eli sellaista opinto-ohjelmaan sisältyvää piiloainesta, jonka olemassaoloa ei edes opetuksesta vastuussa oleva henkilöstö tiedosta.

2.3 Vilppi

Vilpillä tarkoitetaan vilpillistä toimintaa, jossa opiskelija tahallisesti pyrkii vääristämään häneen kohdistuvaa arviointia. Vilpiksi katsotaan myös tilanne, jossa opiskelija jättää tekemättä jonkin opintosuoritukseen kuuluvan työn tekosyyn tai jonkin muun epäpätevän syyn takia.

Vilppiä voivat olla esimerkiksi luntaaminen tentissä, harjoitustyön teettäminen jollain toisella opiskelijalla ja niin edelleen. Dick *et al.* (2002) pitävät toimintaa vilpillisenä jos opiskelijan toiminta rikkoo sääntöjä, jotka on asetettu opiskelijalle, annetulle tehtävälle tai opiskelijan toiminta rikkoo oppilaitoksessa hyväksyttyä yleisesti käytössä olevaa toimintamallia.

Mistä syystä vilppiä pidetään vakavana? Dick *et al.* (2002) esittävät kaksi pääsyitä siihen, miksi vilpin tekemiseen pitäisi aina puuttua. Ensimmäinen syy on se, että opiskelija joka syyllistyy vilppiin ei välttämättä ole saavuttanut sellaista taitotasoa, jota kurssin tai muun opintosuorituksen läpäisemiseksi vaaditaan.

Dick *et al.* (2002) mukaan valmistunut epäpätevä ammattilainen:

- On vaaraksi yhteiskunnalle sillä hän saattaa toimia työssä, jossa hänen virheistään koituu vaara ihmishengille.
- Vahingoittaa ammattinsa arvostusta, koska toimii siinä epäpätevänä.
- Vahingoittaa oppilaitoksen mainetta, koska muut työntekijä saattavat nähdä asian niin, että oppilaitoksesta valmistuvat ovat epäpäteviä.
- Vahingoittaa oman tutkintonsa arvostusta edellisen kohdan syiden perusteella.

Toinen vilppiin puuttumisen syy on Dick *et al.* (2002) mukaan se, että vilppi vahingoittaa opiskelijoita. Edellisten mukaan vilpillinen opiskelija:

- Vahingoittaa oppimisympäristöä. Sekä opiskelijoiden että opettajien tulisi pyrkiä kontrolloimaan vilppiä, jotta saataisiin syntymään positiivisen oppimisen ilmapiiri.
- Vahingoittaa vilppiä tehnyttä opiskelijaa, sillä osa opetetuista asioista on jäänyt oppimatta. Vilpillisyys myös aiheuttaa sen, ettei opiskelija ole täysin valmistautunut työhönsä valmistuttuaan.
- Vahingoittaa kanssaopiskelijoita, jotka eivät toimi vilpillisesti. Vilpillinen opiskelija saa epäreilua etua suhteessa opiskelijoihin, jotka hoitavat tehtävänsä tunnollisesti. Jos arvosanoilla on erityistä merkitystä apurahojen, stipendien ja työpaikan saamisen suhteen, vilpillisellä toiminnalla voi olla vakaviakin seurauksia.

Vilppi on kohtalaisen yleistä. Dick *et al.* (2002) esittelevät kaksitoista tutkimusta, joissa on tutkittu pääosin yliopisto-opiskelijoiden vilpillistä toimintaa. Näistä tutkimusten keskiarvona voidaan laskea, että peräti 75 % yliopisto-opiskelijoista syyllistyy vilppiin opintojensa aikana. Bushweller (1999) raportoi tutkimuksesta, jonka mukaan 80 % USA:n parhaista lukio-opiskelijoista myöntää syyllistyneensä vippiin. Merkittävämpää on kuitenkin, että samassa tutkimuksessa vilpistä ilmoitti jääneensä kiinni vain 5 % vilppiin syyllistyneistä.

2.4 Arvioinnin suhde opetussuunnitelmaan

Varsinkin ensimmäisen ja toisen asteen opinnot suunnitellaan yleensä kansallisesti. Koska kouluilla ja oppilaitoksilla on yhtenäiset opetussuunnitelmat, myös opiskelijoiden menestymistä opinnoissa voidaan mitata kansallisella tasolla samankaltaisin keinoin (Committee on the Foundations of Assessment, National Research Council, 2001).

Joensuun yliopiston tietojenkäsittelytieteen opintojen sisällöt on suunniteltu ACM Computing Curricula 2001 (The Joint Task Force on Computing Curricula, 2001) mukaan. Kukin kurssi on laadittu siten, että opinnot noudattavat ACM:n Computing Curricula -työryhmän suositteleman yleisen tietojenkäsittelytieteen opinto-ohjelman rakenteita. Koska ViSCoS-opinnot pyritään järjestämään mahdollisimman samankaltaisina kun kontaktiopinnot, myös ViSCoS-opintojen sisältö on suunniteltu ACM:n opinto-ohjelman perusteella.

ViSCoS-opinnoissa voidaan arvioida myös sitä, miten hyvin opetusohjelmassa määritellyt asiat on omaksuttu sen lisäksi, että arvioinnilla mitataan opiskelijan henkilökohtaista menestymistä. Käytännössä ViSCoS-opintojen tarkastelu suhteessa koko ACM Computing Curricula 2001:n ei kuitenkaan ole mielekästä. Tutkinto-opiskelijoille tarjottavat lähiopetuskurssit on suunniteltu ja painotettu ACM:n opinto-ohjelman mukaan, joten vastaavuudeksi riittää, että ViSCoS-kurssit vastaavat mahdollisimman pitkälti kontaktiopiskelijoille tarjottavia kursseja.

2.5 Opiskelijan arviointiin käytettävät menetelmät

Arvioinnin avulla pyritään saamaan selville miten hyvin opiskelija ymmärtää opetetun asian. Se minkälaista tietoa opiskelijan osaamisesta saadaan, riippuu pitkälti siitä millaista arviointitapaa käytetään opiskelijan osaamisen kartoittamiseen (Hricko, 2005). Arvioinnilla voidaan myös vaikuttaa paljolti siihen mitä opiskelija muistaa oppimastaan opintosuorituksen jälkeen (Gibbs ja Simpson, 2004).

Karjalainen ja Kemppainen (1994) korostavat, että arvioinnin pääsyynä ei ole yliopistossa pelkästään kontrolloida opiskelijan osaamista ja määrätä arvosanoja, vaan opettaa tiedettä, tieteellistä ajattelutapaa ja toimintamuotoja. Tässäkin suhteessa perinteisen tentin käyttö tulee asettaa kyseenalaiseksi.

Opiskelijan arviointiin käytettävät menetelmät voidaan kategorioida monella tavoin. Karjalainen (2001) esittelee useita tapoja eritellä arviointia. Yksi mahdollisuus jaotella arviointia on tarkastella, mistä lähtökohdasta arviointiin on ryhdytty. Tällöin arviointi voidaan jakaa *sisältyvään arviointiin*, *osallistuvaan arviointiin* sekä *tarkkailevaan arviointiin*.

Näistä ensimmäinen on arvioijan itsensä suorittamaa arviointia, josta käytetään usein termiä itsearviointi. Karjalainen (2001) jatkaa, että osallistuva arviointi sisältyy kaikkien toimintaan, jossa vähintään kaksi yksilöä toimii yhteistyössä. Tällöin ”arvioijana on yhteistyökumppani, arvioitava toimija on yhteistyön toinen osapuoli, arvioinnin kohde on yhteistyön prosessi sekä itse toimintasisältö, arvioinnin intressi on yhteisen toiminnan (tuloksellisuuden) seuranta sekä kehittäminen ja arvioinnin välineet ovat interaktiiviset”. Tarkkailevalla arvioinnilla tarkoitetaan täysin toiminnan ulkopuolelta tapahtuvaa ”observeivaa, kontrolloivaa ja määrittävää arviointia”.

Kirjallisuudessa arviointiin liitetään termit *absoluuttinen arviointi*, *suhteellinen arviointi*, *formatiivinen arviointi*, *summatiivinen arviointi*, *diagnostinen arviointi* sekä *prognostinen arviointi* (Karjalainen, 2001). Absoluuttisella arvioinnilla tarkoitetaan menetelmää, jossa arvioitavat arvioidaan samalla ennalta määrätyllä ja muuttumattomalla kriteeristöllä. Suhteellisessa arvioinnissa arvioitavia vertaillaan toisiinsa ryhmän sisällä. Tällaisessa arviointitavassa koko ryhmän kokonaismenestys vaikuttaa yksittäisten arvioitavien menestymiseen.

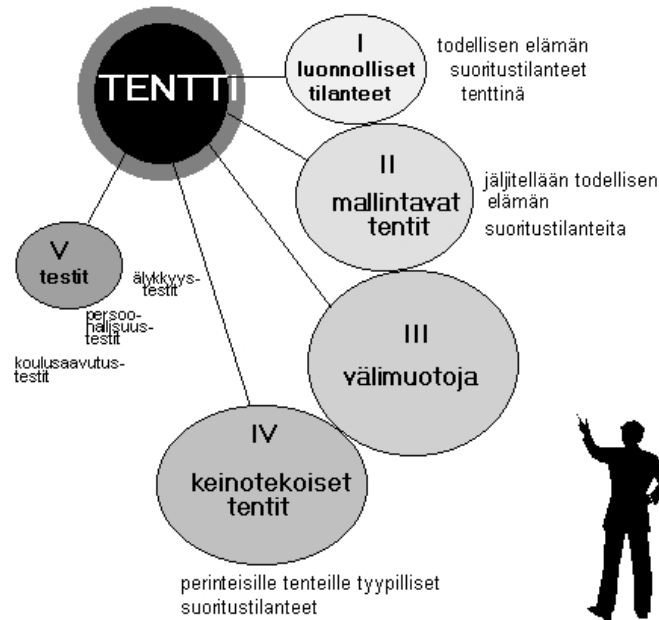
Formatiivisella arvioinnilla tarkoitetaan arviointia, joka tapahtuu varsinaisen opintosuorituksen aikana. Esimerkkejä formatiivisesta arvioinnista ovat esimerkiksi välikokeet ja pistokokeet (Karjalainen, 2001). Summatiivinen arviointi taas tarkoittaa loppuarviointia, joka tapahtuu opintosuorituksen päätteeksi. Yliopistoympäristössä loppukuvailut ovat perinteisiä summatiivisia arviointitilaisuuksia. Diagnostisella arvioinnilla pyritään selvittämään opiskelijan tietoja ja taitoja ennen varsinaisen opetuksen aloittamista. Prognostisella arvioinnilla taas pyritään ennustamaan opiskelijan tulevaa opintomenestystä aikaisempien opintosuoritusten perusteella.

Karjalainen (2001) toteaa, että viime vuosikymmenellä yleisiin arviointikäsitteisiin on otettu mukaan termit *autenttinen arviointi*, *suora arviointi*, *matriisiarviointi*, *sopimusarviointi* sekä *itsearviointi*. Näistä autenttisella arvioinnilla tarkoitetaan ”mittaavan arvioinnin vastakohtana luonnollista ja mahdollisimman todellisen elämän kaltaista arviointia” (Karjalainen, 2001). Suoralla arvioinnilla tarkoitetaan arviointia, jossa arvioinnin välivaiheet on minimoitu. Sen sijaan, että opiskelijan osaamista mitattaisiin paperilla esitetyin kysymyksiin, suorassa arvioinnissa pyritään havainnoimaan opiskelijan menestystä todellisessa toimintatilanteessa.

Matriisiarvioinnissa arviointi pyritään tekemään mahdollisimman läpinäkyväksi laatimalla etukäteen mahdollisimman tarkka matriisin muotoon arviointikriteeristö (Karjalainen, 2001). Tällaisesta arvioinnista on käytetty myös termiä *rubriikkiarviointi*. Käsitellen matriisiarviointia tarkemmin luvussa 4.3. Sopimusarvioinnilla tarkoitetaan arviointia, jossa arvioinnin kriteeristö ja arviointitekniikat sovitaan yhdessä arvioitavan kanssa. Itsearviointissa arvioinnin suorittajana toimii arvioitava. Karjalainen (2001) sijoittaa itsearviointin käsitteen vahvasti autenttiseen arviointiin. Palaan myös itsearviointiin tarkemmin luvussa 4.3.

Karjalainen ja Kempainen (1994) jakavat arviointimenetelmät eli oman terminologiansa mukaan tenttimenetelmät Kuvan 3 mukaisella tavalla. *Luonnollisen tilanteen*

tentit ovat kirjoittajien mukaan arviointitilaisuuksia, joissa opiskelijat toimivat luonnollisessa tilanteessa samaan tapaan, kuin valmistuttuaan. Esimerkkinä on arkkitehtio-piskelijöiden osallistuminen suunnittelukilpailuun omilla töillään.



Kuva 3: Mahdolliset tenttirakenteet (Karjalainen ja Kempainen, 1994)

Mallintavalla tentillä Karjalainen ja Kempainen (1994) tarkoittavat sellaisten asioiden testaamista, joita ei ole turvallista tai mielekästä testata luonnollisissa oloissa vaan simuloimalla aitoa tilannetta. Kirjoittajat jakavat vielä mallintavat tentit produkti- ja prosessisuuntautuneisiin, joista ensimmäisessä keskitytään työn lopputulokseen ja jälkimmäisessä itse työskentelyprosessiin. Esimerkki mallintavasta tentistä on lääketieteen opiskelijoilla hypoteettinen potilas lääkärikäynnillä.

Karjalainen ja Kempainen (1994) käyttävät termiä *keinotekoinen tentti* perinteisen kurssitenttin kaltaisilta ulkoa muistamista mittaavilta tenteiltä. Edellisten mukaan keinotekoiset tentit ovat ”Sellaisia tenttimistä varten rakennettuja tilanteita, joilla ei toimintarakenteidensa puolesta ole vastaavuutta todellisen elämän kanssa”. *Välimuoto-tenteiksi* kirjoittajat määrittelevät sellaiset tentit joissa ”ollaan keinotekoisien ja mallintavien tenttien välimaastossa”. Tähän kategoriaan kuuluvat tentit, joilla on piirteitä sekä keinotekoisista tenteistä, että mallintavista tenteistä. Eräs esimerkki tällaisesta tentistä on käsitekarttatentti, jossa opiskelija laatii käsitekartan annetusta aiheesta.

Viides arviointitapa, jonka Karjalainen ja Kemppainen (1994) määrittelevät on standardoidut tieteelliset testit, kuten kouluavustustestit ja älykkyystestit. Kirjoittajat ovat kuitenkin sitä mieltä, että varsinaisia testejä ei missään tapauksessa tule käyttää opiskelijoiden arviointiin yliopistossa, sillä niillä on yleensä muu tarkoitus. Varsinkin USA:ssa standardoitujen testien käyttäminen arvioinnissa on kuitenkin yleistä.

Sanders ja Horn (1995) esittävät artikkelissaan, että testejä ja testausta vastaan esitetty kritiikki on aiheetonta ja yhtenäisellä testauksella olisi saavutettavissa etuja valtakunnallisesti. Tietojenkäsittelytieteen verkko-opetus on kuitenkin kansallisessa mittakaavassa niin vähäistä, ettei validoitujen standardoitujen testien laatiminen oppimisen arviointiin ole järkevää Suomessa.

Edellä kuvattu jaottelu on tarkoitettu erilaisten tenttimallien teoreettista tarkastelua varten (Karjalainen ja Kemppainen, 1994). Kirjoittajien mukaan käytännön arviointimenetelmää valitessa ei ole välttämätöntä luokitella arviointimenetelmiä edellä kuvatulla tavalla. Tärkeämpää on pyrkiä välttämään arvioinnin kaksoisrakenteisuutta ja valita kurssille sopivin arviointimenetelmä.

2.6 Hyvän arviointimenetelmän ominaisuudet

Arviointimenetelmiä tarkastellessa on tärkeää pohtia sitä, millainen on hyvä arviointimenetelmä. Karjalainen ja Kemppainen (1994) kiteyttävät hyvän yliopistollisen arviointimenetelmän ominaisuudet seuraavasti:

- opettaa etsimään ja löytämään tietoa
- opettaa arvioimaan tietoa
- opettaa soveltamaan tietoa
- opettaa esittämään tietoa ymmärrettävällä tavalla ja keskustelemaan siitä
- opettaa opiskelijaa myöntämään ja korjaamaan omat erehdykset ja tietämättömyyden
- opettaa rakentavasti arvioimaan omaa ja kollegan suoritusta
- opettaa ammatilliseen yhteistyöhön

- ei sisällöltään eikä rakenteeltaan aiheuta opiskelijalle kohtuutonta psyykkistä stressiä ja ahdistusta

Edellisten lisäksi hyvän arviointimenetelmän tulee olla eettisesti kestäväällä pohjalla. Karjalainen (2001) laskee arvioinnin eettisiin perusvaatimuksiin seuraavat ominaisuudet:

- kriteerit ovat julkilausuttuja (eksplisiittisiä) ja arviointi perustuu niihin
- kriteerit ovat rationaalisesti perusteltuja ja loogisia - sisällöllisiä
- kriteerit ovat tiedossa ennen suoritusta, sen aikana ja myös suorituksen jälkeen
- kriteereistä voidaan keskustella argumentatiivisesti ja niihin on mahdollista vaikuttaa järkevin perustein

Opettajan kannalta on tärkeää, että arviointimenetelmä antaa mahdollisimman todennukaisen kuvan opiskelijan osaamisesta ja taitotasosta. Hyvä arviointimenetelmä opettaa myös opettajaa (Karjalainen ja Kemppainen, 1994). Opettajan kannalta arviointiin käytettävän työmäärä voi toimia myös hyvän arviointimenetelmän mittarina. Jos arviointimenetelmä vaatii suuren työpanoksen suhteessa siitä saatavaan hyötyyn, on arviointimenetelmän vaihtaminen perusteltua.

3 Tietojenkäsittelytieteen verkko-opintojen arviointi

ViSCoS-opinnot ovat tietojenkäsittelytieteen korkeakouluopintoja. Opintojen arviointiin käytetään samankaltaisia menetelmiä, kuin muissakin yliopisto-opinnoissa. Arvioinnin kannalta ViSCoS-opintojen verkko-opetusluonne sekä opetettava asiasisältö nousee keskeiseksi arviointia suunniteltaessa.

Carter *et al.* (2003) käsittelevät työryhmänsä raportissa arvioinnin ongelmia ja esittelevät kyselytutkimuksen, jossa tietojenkäsittelytieteen opettajat selvittävät käyttämiään arviointimenetelmiä. Työryhmän suorittaman kyselyn mukaan tietojenkäsittelytieteen opetuksessa käytetyt arviointimenetelmät keskittyvät erityisesti opiskelijan ongelmanratkaisukyvyyn mittaamiseen.

Tietojenkäsittelytieteen ja etenkin ohjelmoinnin oppimisessa onkin pidetty erityisen tärkeinä opiskelijan ominaisuuksina sekä ongelmanratkaisukykyä että loogista päättelykykyä. Tämän voi nähdä myös tarkastelemalla tietojenkäsittelytieteen valtakunnallisia pääsykoekysymyksiä. Vuosittain valtaosa pääsykoekysymyksistä on erityisesti ongelmanratkaisuun painottuvia. Tietojenkäsittelytieteen opetusta järjestävät yliopistot (2008) ovat julkaisseet vanhat pääsykoekysymykset WWW-sivuillaan.

Verkko-opinnoissa ei välttämättä voida hyödyntää kaikkia lähiopetukseen soveltuvia arviointimenetelmiä. Koska opiskelijat suorittavat opintojaan Internetin välityksellä, ei normaalia vuorovaikutusta opettajan ja opiskelijoiden välille perinteisellä tavalla synny. Opiskelijat eivät myöskään kurssin aikana välttämättä ole toistensa kanssa vuorovaikutuksessa. Kurssilla toimivien henkilöiden vuorovaikutussuhteisiin perustuvat arviointitekniikat ovat huomattavasti vaikeampia toteuttaa verkkokurssin puitteissa.

Koska opiskelijat suorittavat opintojaan joko omalta tai yhteistyöoppilaitoksen tietokoneelta, erilaisten ohjelmien asentaminen on erittäin hankalaa. Tämä asettaa omat haasteensa arviointia suunniteltaessa. Etätenttimiseen ja Internetin välityksellä tapahtuvaan arviointiin on kehitetty erilaisia sovelluksia. Tällaisten sovellusten asentamisen vaatiminen on kuitenkin harkittava tarkkaan. Mitä vähemmän ulkopuolisia ohjelmia opiskelijoiden käyttämiltä tietokoneilta vaaditaan sitä parempi.

Tällä hetkellä ViSCoS-opiskelijalla oletetaan minimissään olevan käytössään WWW-selain, ohjelma PDF-tiedostojen lukemiseksi sekä Java-käännösympäristö. Jokainen uusi käyttöön otettava ohjelma lisää kurssien opettajien työtaakkaa. Opettajat

joutuvat ohjaamaan opiskelijoita ohjelmien asennuksessa ja tämä aika on usein pois kurssiin liittyvien asioiden ohjauksesta.

Luvussa 3.1 käsittelen verkko-oppimisympäristön hyödyntämistä arvioinnissa. Vilppi on valitettavan yleistä korkeakouluopinnoissa. Internetissä opetettavilla kursseilla vilpin tekeminen on tutkimusten mukaan vielä yleisempää, kuin perusopetuksessa. Luvussa 3.2 käsittelen vilpin esiintymistä tietojenkäsittelytieteen verkkokursseilla sekä sitä, kuinka vilpin tekemistä voidaan vähentää.

Ohjelmointia pidetään yleisesti eräänä tärkeimmistä tietojenkäsittelytieteen opintojen osa-alueista (The Joint Task Force on Computing Curricula, 2001). Opiskelijoiden oletetaan omaksuvan kelvollinen ohjelmointitaito jo tietojenkäsittelytieteen opintojen alkuvaiheessa. Syventävissä opinnoissa opiskelijoilla oletetaan olevan tiettytasoiset ohjelmointitaidot. Koska ohjelmoinnin oppiminen on haastavaa, aloitteleva opiskelija on suuren haasteen edessä. Luvussa 3.3 käsittelen ohjelmoinnin osaamisen arviointiin liittyviä ongelmia.

Eräs tapa arvioinnin käytännön toteuttamiseksi on tietokoneavusteinen arviointi. Tällaista arviointia on hyödynnetty tietojenkäsittelytieteen opetuksessa erityisesti ohjelmointitehtävien automaattisessa ja puoliautomaattisessa tarkastamisessa. Luvussa 3.4 käsittelen yleisimmät tietokoneavusteisen arvioinnin menetelmät.

Luvussa 3.5 tuon esille verkkotentteihin liittyvän tunnistautumisongelman. On hyvin vaikeaa – ellei jopa mahdotonta – varmistua siitä, että Internetin välityksellä suoritettavaan tenttiin vastaa juuri se henkilö, jonka suoritusta arvioidaan.

3.1 Verkko-oppimisympäristö arvioinnissa

Opintojen arvioinnin tarkastelussa tulee ottaa huomioon myös *oppimisympäristön* käsite. Laajasti oppimisympäristöllä tarkoitetaan sitä kokonaisuutta, jossa opiskelu tapahtuu (Meisalo *et al.*, 2003). Oppimisympäristöön kuuluvat esimerkiksi opettaja, opiskelijat, oppimateriaali, luokkatila ja niin edelleen.

Yleiskielessä oppimisympäristöllä tarkoitetaan usein myös *verkko-oppimisympäristöä* Meisalo *et al.* (2003) mukaan *oppimisalustaa*. Verkko-oppimisympäristöllä ei ole tällä hetkellä ole vielä yhtenäistä määritelmää. Lukkarinen (2003) kokoaa yhteen eri-

laisia verkko-oppimisympäristön määritelmiä ja toteaa, että niille on yhteistä se, että verkko-oppimisympäristöllä tarkoitetaan paikkaa tai tilaa jossa opiskella.

Käytännössä verkko-oppimisympäristöt tarjoavat ainakin työvälineen oppimateriaalin välittämiseen ja erilaisten harjoitustehtävien jakamisen ja palauttamisen järjestämiseen sekä jonkinlaisen keskustelualueen opiskelijoiden ja opettajien käyttöön. Tällaisia verkko-oppimisympäristöjä ovat esimerkiksi Blackboard (Blackboard Inc., 2007), FLE3 (UIAH Media Lab, Taideteollinen korkeakoulu, 2007) sekä avoimen lähdekoodin Moodle (Moodle-yhteisö, 2007).

ViSCoS-opinnot järjestetään nykyisin hyödyntäen verkko-oppimisympäristönä Moodle-ympäristöä. Aikaisemmin käytössä oli kaupallinen WebCT-ympäristö, joka on ViSCoS-Moodlen valinnan jälkeen yhdistynyt Blackboard-ympäristöön. Oppimisympäristön vaihdokseen päädyttiin, sillä Moodle tarjosi avoimena ympäristönä enemmän mahdollisuuksia erilaisten lisäsovellusten ja avoimen ohjelmointirajapintansa ansiosta. Lisäksi ViSCoS-kurssien opettajat kokivat WebCT:n kankeaksi käytännön opetustyössä.

Modernit verkko-oppimisympäristöt tarjoavat nykyisin erilaisia välineitä arvioinnin toteuttamiseksi. Lisäksi oppimisympäristöihin on nykyisin yleensä liitetty mahdollisuus järjestää ainakin jonkinlaisia verkkotenttejä tai monivalintakyselyitä. Myös muita arviointityökaluja on tarjolla useisiin verkko-oppimisympäristöihin. Avoimen lähdekoodin verkko-oppimisympäristöt mahdollistavat tämän lisäksi vapaasti uusien ominaisuuksien lisäämisen ympäristöön. On myös mahdollista laatia itse kokonaan uusia arviointivälineitä käytettäväksi avoimen lähdekoodin oppimisympäristössä.

Verkko-oppimisympäristöt voidaan nähdä arviointivälineinä, mutta toisaalta ne ovat myös alusta erilaisten verkkopohjaisten arviointimenetelmien hyödyntämiseksi. Verkko-oppimisympäristöjä tulee kuitenkin ajatella vain erilaisia arviointitapoja mahdollistavina työkaluina. Verkko-oppimisympäristö ei saisi rajoittaa erilaisten arviointimenetelmien hyödyntämistä, vaan päin vastoin, tarjota joustavampia mahdollisuuksia arvioinnin toteuttamiseksi.

3.2 Vilppi tietojenkäsittelytieteen verkko-opinnoissa

Luvussa 2.3 kävin läpi vilpin ongelmaa yleisesti. Nyt ulotan näkökulman koskemaan tietojenkäsittelytieteen verkko-opintoja. Sheard ja Dick (2003) mukaan vilpin esiintyminen riippuu pitkälti oppiaineesta. Kirjoittajien mukaan muiden oppiaineiden löydökset vilpin esiintymisen todennäköisyydestä eivät välttämättä kerro siitä, mikä tilanne on tietojenkäsittelytieteen opiskelijoiden osalta. Sheard *et al.* (2003) raportoivat kuitenkin myös omassa tietojenkäsittelytieteen opiskelijoita koskevassa tutkimuksessaan, että 80 % tutkimukseen osallistuneista opiskelijoista myöntää syyllistyneensä vilpilliseen toimintaan. Lipson ja McGavern (1993) selvittivät kyselytutkimuksessaan, että 50 % opiskelijoista pitää vilpin tekemistä todennäköisempänä jos kyse on ohjelmoinnin opiskelusta.

Dick *et al.* (2002) toteuttamassa tutkimuksessa suuri joukko tietojenkäsittelytieteen opettajia pitää vilpin lisääntymisen yhtenä pääsyyinä kopioinnin helppoutta. Internetissä on ammennettavissa suuri määrä informaatiota, jota opiskelijoiden on mahdollista hyödyntää myös vilpillisesti. Rowe (2004) raportoi kahdesta tutkimuksesta, joiden valossa näyttää siltä, että mitä vähemmän informaatiota liikkuu arvioijan ja arvioitavan välillä, sitä todennäköisemmäksi vilpin tekeminen tulee. Käytännössä tämä tarkoittaa kirjoittajan mukaan sitä, että opiskelijat, jotka tuntevat olevansa ”etäämpänä”, syyllistyvät todennäköisemmin vilpin tekemiseen. Koska etäopetuksessa vuorovaikutusta opettajan kanssa on vähemmän kuin perinteisessä luokkaopetuksessa, tämä saattaa johtaa vilpin lisääntymiseen.

Simon (2005) kuvaa artikkelissaan vesileimajärjestelmän, jota käytettiin selvittämään vastasivatko opiskelijat tietojenkäsittelytieteen verkkotentissä henkilökohtaisesti tenttikysymyksiin. Kirjoittajan kuvaamassa ympäristössä opiskelijoilla oli 48 tuntia aikaa ladata tenttikysymykset. Kun opiskelija oli ladannut vastaukset, hänen tuli vastata kysymyksiin ja palauttaa tietyn ajan kuluessa tiedosto, jossa kysymykset ja vastaukset olivat. Järjestelmässä kysymykset oli vesileimattu. Kunkin tentin kysymykset oli laadittu värillä, joka näytöllä ja tulostettuna vaikutti mustalta, mutta joka oli tentikohtainen.

Simon (2005) onnistui järjestelmän ensimmäisellä käyttökerralla paljastamaan useita vilpin yrityksiä. Kirjoittajan mukaan ilmeni tapauksia, joissa opiskelijat palauttivat vastaustiedostoja, jotka oli ladattu järjestelmästä eri aikaan kun heille yksilöidyt tentit. Artikkelista ei kuitenkaan käy ilmi miten monta opiskelijaa osallistui kurssille ja miten moni opiskelija jäi kiinni vilpillisestä toiminnasta. Kirjoittajan mukaan järjestelmä ei

toiminut enää yhtä hyvin myöhemmillä testauskerroilla. Simon selittää tämän sillä, että vilppiä kyllä tapahtui edelleen, mutta opiskelijat olivat oppineet minkälaisesta vilpistä kurssilla jäi kiinni.

Toistaiseksi ViSCoS-opinnoissa ei ole käytetty verkkotenttejä. Koska etäopetuksessa opettajalla ei ole mahdollisuutta tavata opiskelijaa kasvotusten, opettajan on mahdollista tietää kuka verkon välityksellä opintoja suorittaa ja palauttaa vastauksia annettuihin kysymyksiin tai harjoituksiin. Tähän ongelmaan palaan tarkemmin luvussa 3.5.

Eri kulttuureista lähtöisin olevat opiskelijat suhtautuvat vilppiin hyvin eri tavoin (Flynn, 2003). Tämä tulee esille myös Dick *et al.* (2002) toteuttamassa kyselytutkimuksessa. Erityisesti Aasiasta kotoisin olevat opiskelijat syyllistyvät kyselyyn osallistuneiden tietojenkäsittelytieteen opettajien mukaan vilppiin huomattavasti paikallisia opiskelijoita useammin. Kulttuurierot suhtautumisessa vilppiin saattavat muodostua ongelmaksi myös kansainvälistyvässä ViSCoS-opinnoissa ja tämä tulee voida ottaa huomioon opintoja suunniteltaessa.

Eräs yleisimmistä vilpillisistä opiskelumenetelmistä on kopioida tekstiä joko Internetistä tai jostain muusta lähteestä viittaamatta tekstin alkuperäiseen kirjoittajaan. Tällaista toimintaa kutsutaan *plagiarismiksi*. Vakavimmillaan tämä tarkoittaa kokonaisen esseen, ohjelman tai muun arvioitavan työn täydellistä kopioimista ilmoittamatta alkuperäistä lähdettä. Lukashenko *et al.* (2007) esittelevät tunnetuimpia plagiarismin tunnistamiseksi käytettäviä pääasiassa kaupallisia työkaluja sekä niiden nykytilaa. Kirjoittajat mainitsevat työkalujen suurimmaksi ongelmaksi sen, että ne eivät kunnolla tunnista sitä, onko kyseessä todellinen plagiointi vai oikein viitattu teksti.

Lukashenko *et al.* (2007) pitävät plagioinnin tunnistamiseen käytettäviä automaattisia työkaluja käyttökelpoisina, mutta toteavat, että tekstin lukeva opettaja on kuitenkin huomattavasti pätevämpi tunnistamaan plagioinnin tekstistä. Monet automaattiset ohjelmointivastausten tarkistimet sisältävät myös plagioinnin tunnistamiseen tarkoitettuja ominaisuuksia (ks. luku 3.3). Plagioinnin tunnistavien ohjelmien liittäminen osaksi Moodle-ympäristöä on toteutettavissa kohtuullisen helposti.

Vilpin esiintyminen on havaittu myös ViSCoS-ympäristössä. Havaitut ongelmat koskevat ViSCoS:ssa erityisesti plagiointia. Ajoittain osa esseistä on kopioitu Internetistä. Toistaiseksi opiskelijoiden yhteistyötä kurssin harjoitustehtävien tekemiseksi ei ole pi-

detty plagiointina vaan työskentelytapana. Sen sijaan laajempien harjoitustöiden osalta plagiointiin on suhtauduttu vakavasti.

3.2.1 Vilpin vähentäminen

Karjalainen ja Kemppainen (1994) esittävät hyvin erilaisen näkemyksen vilpistä tenttitilanteissa. Heidän mukaansa vilpillisyys pitäisi hyväksyä ja hyödyntää tenttitilanteissa koska yliopisto-opettajat suhtautuvat perinteisesti liian negatiivisesti opiskelijoiden moraaliin opintosuorituksia koskevilla asioilla. Kirjoittajien mukaan ”tosiasiasa opiskelijat toimivat erittäin vastuullisesti, jos heille vain annetaan siihen mahdollisuus”.

Oppimistilanteeksi rakentuvassa tentissä ei nähdäksemme voi tehdä vilppiä perinteisessä mielessä. Jotta epäselvyyksiä ei tulisi, on opettajan ja opiskelijan aina yhdessä sovittava pelisäännöistä. Tenttimallista keskusteltaessa — mielellään heti luennon alussa — tulee sopia hyväksytyt työmuodot ja toimintatavat (Karjalainen ja Kemppainen, 1994).

Vilpin esiintyminen erityisesti verkko-opinnoissa näyttää tutkimusten perusteella olevan valitettavan yleistä. Vaikka Karjalainen ja Kemppainen (1994) mainitsevatkin tehokkaimmaksi vilpin estämiskeinoksi tenttien ja arvioinnin muuttamista, sellaiseksi ettei vilppiä voi esiintyä, tämä tuskin on käytännössä mahdollista tietojenkäsittelytieteen verkko-opinnoissa. ViSCoS-opintojen oppimisen arviointi tuleekin järjestää niin, että vilpin esiintymisen mahdollisuuteen varaudutaan ja opiskelijan vilpillinen suoritus pystytään tunnistamaan kohtuullisella vaivalla.

Sheard *et al.* (2003) laativat kaksiosaisen tutkimuksen, jossa he selvittivät tietojenkäsittelytieteen opiskelijoiden syyllistymistä vilppiin sekä syitä, jotka johtivat vilpilliseen toimintaan. Tutkimuksen ensimmäisessä osassa, jossa selvitettiin vilpin yleisyyttä, kävi ilmi, että suurimmat syyt vilpilliseen toimintaan ovat kiireisen aikataulun aiheuttama paine ja tehtävien vaikeus. Opiskelijoiden vastauksista voitiin päätellä, että tehokkain tapa vähentää vilpin esiintymistä on saada opiskelijat ottamaan vastuuta omasta työskentelystään. Kirjoittajien mukaan vastaavia tuloksia on saatu aikaisemmista aiheita koskevista tutkimuksista.

Tutkimuksen toisessa osassa Sheard *et al.* (2003) selvittivät tarkemmin tietojenkäsittelytieteen opiskelijoiden opiskelukokemuksia ja opiskeluun liittyviä ongelmia. Tutkimuksessa ei varsinaisesti tutkittu vilpin esiintymistä, vaan niitä asioita opiskelijat kokiivat opintoja vaikeuttaviksi tekijöiksi. Edellisen tutkimuksen valossa kirjoittajat päättelivät näiden tekijöiden olevan vilpilliseen toimintaan johtavia syitä. Heidän mukaansa vastauksista näkyviä pääosin opiskelijoista lähtöisin olevia syitä vilpin esiintymiseen ovat opiskelijoiden oma ajanhallinta, puutteellinen valmistautuminen opintosuoritukseen, haluttomuus noudattaa hyväksi havaittuja käytäntöjä, avun pyytämiseen liittyvä suuri kynnys sekä vähäinen kiinnostus opiskeltavaan aiheeseen.

Opiskelijasta riippumattomiksi syiksi Sheard *et al.* (2003) nostivat laitteisto-ongelmat, ohjelmisto-ongelmat, ohjelmoinnin oppimisen kumulatiivisen luonteen, soveltuvien tietolähteiden puutteen, huonosti toteutetut harjoitustilanteet, huonosti laaditut tehtävät, liian helposti saatavissa olevat valmiit vastaukset, ryhmätyöt sekä muiden oppiaineiden aiheuttamat ongelmat. Ohjelmoinnin kumulatiivisella luonteella tarkoitetaan tässä yhteydessä sitä, että uusi tieto rakentuu vanhan pohjalle. Mikäli opiskelija pitää väliviikon luennoista ja harjoituksista, hänen on usein mahdotonta seurata opetusta opiskelematta ensin itsenäisesti aikaisemmin käsitellyt asiat.

Vilpin esiintyminen, siihen suhtautuminen, sekä erityisesti sen torjunta on tärkeää pohdittava hyvin tarkasti opintoja suunniteltaessa ja toteutettaessa. Erityisesti ViSCoS-opintojen uskottavuus riippuu pitkälti siitä, miten varmoja voidaan olla siitä, että opiskelijat ovat todella omaksuneet kursseilla käsitellyt asiat. Tutkimusta siitä, miten yleistä vilppi on ViSCoS-opinnoissa ja mitkä ovat siihen johtavia syitä, ei toistaiseksi ole tehty. Edellä olevien tutkimusten valossa on erittäin tärkeää saada opiskelija kokemaan itsensä vastuulliseksi omista opinnoistaan vilpin esiintymisen vähentämiseksi.

3.3 Ohjelmoinnin osaamisen arvioinnin erityishaasteet

Ohjelmoinnin oppiminen on hyvin haasteellista ja siksi myöskään ohjelmoinnin opettaminen ei ole helppoa. Ohjelmoinnin oppiminen (paradigmasta ja ohjelmointikielestä riippumatta) vaatii ns. *algoritmisen ajattelun* oppimista. Algoritmisella ajattelulla tarkoitetaan yleisesti kykyä hahmottaa ongelma sekä menetelmä muodostaa ongelman algoritmisen ratkaisumalli. Ohjelmointia lähdetään kuitenkin yleensä opiskelemaan jon-

kin tietyn ohjelmointikielen avulla. Fincher (1999) esittää, että ohjelmoinnin opettaminen tulisikin aloittaa syntaksiriippumattomasti.

Ohjelmointikieltä opiskeltaessa myös syntaksi tulee opetella ja tämä vaikeuttaa varsinaista algoritmisen ajattelun oppimista (Fincher, 1999). Syntaksissa olevat virheet saattavat viedä opiskelijan huomiota ongelman ratkaisutavasta ja varsinaisesta algoritmista. Ben-Ari (2001) esittää ratkaisuksi *konstruktivistiseen oppimisteoriaan* pohjautuvien opetusmenetelmien hyödyntämisen tietojenkäsittelytieteen opinnoissa. Kirjoittajan mukaan opiskelijan täytyy ensin muodostaa mentaalinen malli tietokoneesta ja sen sisäisestä toiminnasta, ennen kun ohjelmoinnin oppiminen on mahdollista.

Burton ja Bruhn (2003) pitävät erittäin tärkeänä sitä, että opiskelijalla on riittävä matemaattinen pohjatieto ohjelmoinnin oppimiseksi. Ohjelmointikielestä riippuen funktiot, aliohjelmat tai menetöt muistuttavat pitkälti matemaattisia funktioita syötteineen ja tuloksineen. Lisäksi algoritmien laatiminen, muuttujan käsitteen ymmärtäminen ja algoritmien tehokkuuden vertailu on helpompaa, mikäli opiskelija on jo oppinut matematiikkaa opiskellessaan soveltamaan vastaavia asioita.

Ohjelmoinnin psykologiaa käsittelevässä uraa uurtavassa tutkimuksessaan Sheil (1981) päätyi siihen johtopäätökseen, että ohjelmoiminen on ennen kaikkea opittu taito. Toisin sanoen ohjelmoimaan oppii ainoastaan ohjelmoimalla ja harjaantumalla. Mead *et al.* (2006) kokoavat yhteen 25 vuoden ajalta tutkimuksia ohjelmoinnin opettamisesta ja toteavat, että kaikkien tutkimusten tekijät ovat päätyneet samaan johtopäätökseen: ohjelmoimaan oppiminen on hankalaa useimmille opiskelijoille.

Robins *et al.* (2003) tarkastelevat artikkelissaan ohjelmoinnin psykologian tutkimusta noviisien näkökulmasta. Myös he päätyvät johtopäätökseen, että ohjelmoinnin oppiminen on vaikeata, koska siihen liittyy kompleksisen uuden tiedon hankkimista, oikeanlaisten strategioiden omaksumista sekä käytännön taitoja, joiden saamiseksi opiskelijan pitää harjoitella runsaasti. Ohjelmointia pidetään kuitenkin yleisesti eräänä tärkeimmistä tietojenkäsittelytieteen opintojen osa-alueista (The Joint Task Force on Computing Curricula, 2001).

Torvinen (2001) on tutkinut ViSCoS-opiskelijoiden ohjelmoinnin oppimiseen liittyviä ongelmia. Eräs tärkeimmistä tutkimuksesta esiin nousevista seikoista on ohjauksen merkitys ohjelmoinnin oppimisessa. Tämän lisäksi verkko-opiskelu koettiin vieraaksi. Tutkimuksen aikana opintoja suorittivat vain lukiolaiset. Samankaltaiset ongelmat kui-

tenkin ovat havaittavissa myös opiskelijakannan monimuotoistumisen jälkeenkin. Tähän viittaavia tuloksia ohjelmoinnin oppimisen ongelmista on saanut Pölönen (2005). Kirjoittaja onkin tutkinut erityisesti opiskelijan saaman ohjauksen vaikutusta opintoihin ohjelmointikursseilla. Hyvin tärkeäksi Pölönen nostaa palautteen opiskelijan suoriutumisesta. Hän myös muistuttaa, että opiskelijat kaipaisivat erityisesti välitöntä palautetta harjoitustehtävistään. Tällaisen palautteen antaminen onnistuu ainoastaan automaattista arviointia käyttämällä.

Myös ohjelmoinnin osaamisen arviointi on haasteellista. ViSCoS-opinnoissa ohjelmointikurssien arviointimenetelmänä käytetään tällä hetkellä perinteisiä tenttejä. Suurimpana osana tenttikysymyksistä on ohjelman tai ohjelman osan laatiminen. Tenttiin vastataan kynällä, kun taas kurssien aikana opiskelijat harjoittelevat ohjelmointia pelkästään tietokoneen avulla. Ohjelman muodostaminen tekstieditorilla on huomattavasti helpompaa sillä teksti on vapaasti muokattavissa. Tietokoneella ohjelmoitaessa pyyhkekumia ei tarvitse käyttää, jotta esimerkiksi rivejä voi lisätä toisten väliin.

Todellisessa käytännön ohjelmointitilanteessa opiskelijalla on aina käytössään kääntäjä sekä mahdollisuus tarkistaa onko ohjelmakoodi syntaktisesti oikein laadittu. Lisäksi hän voi tarkistaa ohjelman toimivuuden suorittamalla sen ja tehdä muutoksia myös ohjelman loogiseen toimintaan testiajon perusteella. Nykyinen ViSCoS-opintojen ohjelmointitentti poikkeaa täysin normaalista ohjelmointitilanteesta. Tällaisessa ohjelmoinnin osaamisen arvioinnissa onkin nähtävissä hyvin voimakas kaksoisrakenne.

Ohjelmaa laatiessaan opiskelija soveltaa oppimaansa ohjelmointikokemusta täysin uuden tuntemattoman ongelman ratkaisussa. Mikäli opiskelija kokee tenttitilaisuuden ahdistavaksi tai pelkää, että annettu aika ei riitä riittävän hyvän vastauksen laatimiseen, opiskelija ei voi tentissä osoittaa osaamistaan. ViSCoS-ohjelmointitentit muistuttavatkin nykyisin enemmän testejä, joissa mitataan opiskelijan kykyä muuntaa ohjelmointitaitonsa taidoksi laatia ohjelman osia ennalta harjoittelematta uudessa ympäristössä. Tentissä tuttuja työvälineitä ei ole käytössä ja tekstieditori on vaihdettu lyijykynään.

Myös kynällä paperille ”ohjelmoitujen” vastausten arvosteleminen on opettajalle hankalaa. Usein opettaja joutuu pohtimaan sitä, onko opiskelija tehnyt virheitä syntaksissa tai ohjelman logiikassa vai johtuvatko virheet täysin vieraasta tavasta laatia ohjelmaa. Eräs vaihtoehto on myös se, ettei opiskelija ole vielä riittävän harjaantunut ohjelmoinnissa. Erityisen ongelmalliseksi tämä muodostuu jos samaa tenttiä tarkastaa useampi

opettaja. Täysin yhdenmukaisten arvosteluperusteiden laatiminen jokaiseen tehtävään on hyvin työlästä ja yksittäisten opettajien arviointi on joka tapauksessa yksilöllistä.

Epätäydellistä vastausta ohjelmointiongelmaan on myös hankala arvioida objektiivisesti. Se, miten kaukana opiskelijan laatima ohjelma on toimivasta ratkaisusta, riippuu täysin opettajan subjektiivisesta arviosta. Tässä tapauksessa ohjelmointivastausten arviointi muistuttaa hyvin pitkälti esseiden arviointia. Opettaja voi toki laatia tiettyjä kriteerejä, joiden perusteella opiskelijan laatima vastaus arvioidaan, mutta viimekädessä kokonaisarviointi perustuu kuitenkin opettajan henkilökohtaiseen näkemykseen opiskelijan laatiman ohjelman laadusta. Ohjelmointitehtävien automaattiseen arviointiin keskityn luvussa 3.4.3.

3.4 Tietokoneavusteinen arviointi

Tietokoneavusteiseksi arviointitekniikka⁴ nimitetään arviointitekniikkaa, jossa ainakin osa opiskelijan työstä arvostellaan automaattisesti tietokonetta hyödyntäen. Tietojenkäsittelytieteen opetuksessa tietokoneavusteisen arvioinnin käyttäminen on Carter *et al.* (2003) järjestämän kyselytutkimuksen mukaan hyvin yleistä. Kirjoittajat selittävät tämän sillä, että tietojenkäsittelytieteen opettajat voivat itse laatia kurssilleen sopivia arviointityökaluja ja tietyn tyyppiset vastaukset on kohtuullisen helppo tarkastaa ohjelmallisesti.

Carter *et al.* (2003) teettämässä kyselyssä tietojenkäsittelytieteen opettajat pitivät seuraavia asioita tärkeimpinä tietokoneavusteisen arvioinnin positiivisina ominaisuuksina:

- Arvioinnissa säästetty aika erityisesti, kun opiskelijaryhmät ovat suuria
- Välitön — joskaan ei välttämättä kovin laadukas — palaute opiskelijalle
- Arvioinnin objektiivisuus ja arviointilinjan yhtenäisyys
- Tietokoneavusteinen arviointi vapauttaa opiskelijat etenemään omassa tahdissaan

⁴Englanniksi Computer Aided Assessment, eli CAA

Carter *et al.* (2003) mukaan samassa kyselyssä opettajien keskuudessa suurimpana ongelmana koettiin se, että tietokonelaitteistoon liittyvät epävarmuudet vaikuttavat arviointiin. Erityisen haitallisiksi nähtiin ajat, jolloin vastauksia analysoiva järjestelmä ei syystä tai toisesta ole käytössä, sekä häiriöt fyysisen tietokonelaitteiston toiminnassa, joiden seurauksena osa arviointitiedosta voi hävitä tai muuttua vääristytyään käyttökelvottomaksi. Myös vilpin tekeminen oli kyselyyn vastanneiden mielestä ongelma, sillä ohjelmallisesti on mahdotonta tarkistaa kuka opiskelijan vastauksen on todellisuudessa syöttänyt järjestelmään.

Carter *et al.* (2003) jakavat tietokoneavusteisen arvioinnin menetelmät viiteen pääkategoriaan:

- Monivalintakysymykset
- Tekstivastaukset eli esseevastaukset
- Ohjelmointitehtävät,
- Visuaaliset vastaukset
- Vertaisarviointi

Käsittelen näiden kategorioiden valossa seuraavaksi kunkin tietokoneavusteisen arvioinnin menetelmää omassa alakohdassaan.

3.4.1 Monivalintakysymykset ja lyhyet tekstivastaukset

Monivalintatehtävissä on useita kysymyksiä, jossa jokaisessa on useampia vastausvaihtoehtoja (Carter *et al.*, 2003). Tällaisissa kysymyksissä on olemassa useita tehtävätyyppejä, kuten esimerkiksi yhdistelytehtävä, aukkotehtävä tai monivalintatehtävä. Opiskelija valitsee tarjolla olevista vastausvaihtoehdoista mielestään oikean vastauksen. Tällaisia tehtäviä voi helposti laatia kaikissa yleisimmissä verkko-oppimisympäristöissä.

Hyvien monivalintakysymysten laatiminen on usein haastavaa (Karjalainen ja Kempainen, 1994). Lister (2001) ottaa kuitenkin voimakkaasti kantaa niiden käytön puolesta ohjelmointikurssien arvostelussa. Käsittelen monivalintakysymysten problematiikkaa tarkemmin luvussa 4.3.

Tekstimuotoisten vastausten tarkistamiseen on olemassa useita lähestymistapoja. Yksinkertaisia tekstikenttiä voidaan tarkastaa esimerkiksi säännöllisillä lausekkeilla tai tarkistamalla ovatko tekstit samanlaisia (Carter *et al.*, 2003). Yksinkertaisten tekstikenttien tarkastaminen on kuitenkin niin triviaalia, että sivuutan sen tarkemman käsittelyn tässä työssä.

3.4.2 Tietokoneavusteinen esseiden arviointi

Kokonaisten esseiden tarkastamiseksi on kehitetty erilaisia tietokoneavusteisia menetelmiä. Valentia *et al.* (2003) vertailevat olemassa olevia esseiden arviointiin käytettäviä järjestelmiä ja toteavat, että ne eivät vielä ole keskenään kovinkaan vertailukelpoisia. Suurin ongelma on yleisessä käytössä olevan esseenäytteiden puuttuminen. Tällainen asiantuntijoiden arvioima esseiden tietokanta, eli ns. *korpus*, helpottaisi huomattavasti erilaisten automaattisten esseentarkastusjärjestelmien keskinäistä vertailua.

Esseentarkistimia kehittävät tutkimusryhmien raporteista on kuitenkin nähtävissä, että nykyiset järjestelmät päätyvät opettajan kanssa 80-90 % todennäköisyydellä samaan arvosanaan (Valentia *et al.*, 2003). Erään suomenkielisten esseiden tarkastamiseksi kehitetyn menetelmän esittelevät Kakkonen ja Sutinen (2004).

Nykyisten automaattisten esseevastausten tarkastimien suurimpana ongelmana on, että ne täytyy opettaa kohtuullisen laajalla aineistolla, ennen kun niitä voi hyödyntää (Valentia *et al.*, 2003). Käytännössä tämä tarkoittaa sitä, että jokaista kysymystä varten pitää olla käytettävissä joukko valmiiksi arvosteltuja vastauksia, joilla järjestelmää on opetettu. Tämän vuoksi automaattista kirjoitelmien arvostelemista ei voi kovinkaan helposti hyödyntää ViSCoS-opintojen arvioinnissa.

3.4.3 Tietokoneavusteinen ohjelmointitehtävien arviointi

Ohjelmointitaidon mittaamiseksi ohjelmallisesti on kehitetty hyvinkin erilaisia keinoja. Koska ohjelmointia pidetään hyvin tärkeänä tietojenkäsittelytieteen opetuksen osa-alueena (ks. luku 3.3) ja opiskelijoiden laatimien ohjelmien tarkastaminen on työllästä, ohjelmien automaattiseen ja puoliautomaattiseen arviointiin on kehitetty useita työvälineitä.

Douce *et al.* (2005a) kokoavat artikkelissaan yhteen automaattisten ja puoliautomaattisten ohjelmointivastausten arviointiin ja arvosteluun kehitettyjen työkalujen kehitystä. Kirjoittajien mukaan ensimmäisen sukupolven tarkastimet kehitettiin 1960-luvulla ja ne oli sidottu opetusta antavien yliopistojen ja koulujen palvelimien käännös- ja ohjelmointiympäristöihin.

Toisen sukupolven järjestelmiä Douce *et al.* (2005a) nimittävät työkalulähtöisiksi järjestelmiksi. Tässä vaiheessa alettiin järjestelmien avulla arvioida ohjelman toimivuuden lisäksi opiskelijan ohjelmointityyliä. Erotuksena ensimmäisen sukupolven järjestelmistä, toisen sukupolven järjestelmät oli selkeästi kehitetty tarkoitukseensa sopivaksi, eikä niissä enää hyödynnetty pelkästään valmiiksi palvelimilla olevia ohjelmistoja. Esimerkkejä toisen polven järjestelmistä ovat ASSYST (Jackson ja Usher, 1997), BOSS (Joy ja Luck, 1998), Ceilidh (Higgins *et al.*, 2003) sekä Scheme-Robo (Saikkonen *et al.*, 2001).

Kolmannen polven järjestelmiksi Douce *et al.* (2005a) luokittelevat järjestelmät, jotka hyödyntävät Internetiä sekä tuovat arviointiin uusia kehittyneempiä menetelmiä. Näistä kolmannen ja tuoreimman sukupolven järjestelmistä kirjoittajat nostavat esiin Ceilidh-järjestelmän seuraajan CourseMarkerin (Higgins *et al.*, 2003), kehittyneemmän version BOSS-järjestelmästä (Joy *et al.*, 2005), RoboProf-järjestelmän (Daly ja Waldron, 2004) sekä ASAP-järjestelmän (Douce *et al.*, 2005b).

Kolmannen polven ohjelmointijärjestelmät ovat ViSCoS-opintojen kannalta kaikkein kiinnostavimpia, sillä ne soveltuvat huomattavasti paremmin verkko-opinnoissa käytettäväksi. Lehtonen (2002) tutki pro gradu -työssään CourseMarkerin ja BOSS-järjestelmän soveltuvuutta ViSCoS-opintoihin. Koska tutkielma on vuodelta 2002, ViSCoS-kurssit ja BOSS- sekä CourseMarker-työkalut ovat jonkin verran kehittyneet tutkielman valmistumisen jälkeen. Pääosin CourseMarkerin ja BOSS-ohjelmiston toiminta on kuitenkin pysynyt hyvin samankaltaisena aikaisempiin versioihin nähden muun muassa tehtävien laatimisen kannalta. Tästä syystä Lehtosen päätelmät ovat edelleen käyttökelpoisia tarkasteltaessa sitä, soveltuvatko nämä järjestelmät nykyisiin ViSCoS-opintoihin.

Lehtonen (2002) toteaa tutkielmansa johtopäätöksissä, että molemmilla järjestelmillä harjoitustehtävien laatiminen on monimutkainen ja aikaavievä prosessi. Lisäksi Lehtonen kuvaa molempien järjestelmien asentamista ja omaksumista hyvin hankalaksi. Uusien harjoitustehtävien laadinta on erittäin työlästä ja myös muutosten tekeminen

harjoitustehtäviin on hankalaa. Kirjoittaja myös mainitsee, että tarkkaan määriteltävä syöte ja tulostusmuoto rajaisivat pois käytöstä suuren osan nykyisin ViSCoS-kursseilla käytetyistä harjoitustehtävistä.

CourseMarker- ja BOSS-järjestelmillä voidaan arvostella ainoastaan tekstitulosteita tuottavia ohjelmointivastauksia. Tämä rajaa kokonaan pois graafisten käyttöliittymäkomponenttien sekä applettien käytön harjoitustehtävissä. Molemmat järjestelmät tarvitsevat lisäksi toimiakseen opiskelijan tietokoneelle asennettavan erillisen asiakasohjelman, jolla vastaukset lähetetään palvelimelle (Lehtonen, 2002). Tähtäkään osin järjestelmät eivät ole muuttuneet Lehtosen tutkielman valmistuttua.

Lehtonen (2002) suosittelee kuitenkin tutkielmansa lopussa uusia käytännön testejä sekä CourseMarker- että BOSS-järjestelmillä, sekä muutamilla muilla ohjelmistoilla, jotka hän mainitsee tutkielmassaan. Valitettavasti tällaisia kokeita ei ole ViSCoS:n puitteissa järjestetty. Tämä on kuitenkin varsin ymmärrettävää, sillä kuten kirjoittaja itsekin mainitsee, järjestelmien käyttöä ei opi muutamassa päivässä.

Eräs Douce *et al.* (2005a) luettelemista kolmannen sukupolven ohjelmointitehtävien tarkastamiseen suunnitelluista järjestelmistä on ASAP (Douce *et al.*, 2005b). Järjestelmä on kuitenkin vielä kehityksen alla, eikä sen käytöstä ole kokemuksia muualla, kun kehittäjäyliopistossa. RoboProf-ympäristö (Daly, 1999) taas on kehitetty enemmänkin oppimateriaalin omaksumista tukevaksi ympäristöksi. RoboProf-ympäristössä opiskelijalle tarjotut tehtävät monimutkaistuvat sitä mukaa kun opiskelijan taidot karttuvat. Osa järjestelmän esittämistä kysymyksistä on monivalintatehtäviä. Samankaltaiseen ajatukseen perustuu Suomessa kehitetty Javala-järjestelmä (Lehtonen, 2006). Javalan harjoitustehtäviä on tarjottu ViSCoS-opiskelijoille vapaaehtoisesti suoritettavana lisämateriaalina.

Jackson (2000) esittää näkemyksen, että ohjelmointitehtävien vastausten arvioinnin ei tulisi koskaan olla täysin automaattista, vaan ohjelmistoja voidaan käyttää ainoastaan arvioinnin apuna. Myös BOSS-järjestelmässä lopullisen arvosanan antaa ihminen perehdyttyään BOSS:n laatimaan analyysiin opiskelijan vastauksesta (Lehtonen, 2002). CourseMarker sen sijaan arvostelee opiskelijoiden vastaukset tarvitsematta ihmisen apua arviointiin.

Palaan vielä tarkemmin ohjelmointitehtävien automaattiseen ja puoliautomaattiseen tarkastukseen laadittujen ohjelmien hyödyntämiseen ViSCoS-opintojen ohjelmointikursseilla luvussa 6.4.

3.4.4 Visuaalisten vastausten tietokoneavusteinen arviointi

Visualisointeja voidaan käyttää hyödyksi arvioitaessa esimerkiksi opiskelijoiden ymmärrystä tietorakenteista sekä algoritmeista (Carter *et al.*, 2003). Erään esimerkin tällaisesta arviointiin sopivasta työkalusta esittelevät Malmi *et al.* (2002). Kyseessä on TRAKLA-järjestelmä, joka on suunniteltu algoritmien simuloimisen ja hahmottamisen välineeksi. Carter *et al.* (2003) määrittelevät tähän kategoriaan kuuluviksi myös erilaiset vuokaavioiden ja suunnittelukaavioiden automaattiseen tarkastamiseen suunnitellut ohjelmat.

Visualisointityökaluja on käytetty hyväksi ViSCoS-kursseilla lähinnä opetuksen tukimateriaalina. Esimerkiksi Jeliot-järjestelmän (Kannusmäki *et al.*, 2004) käyttöä on tarjottu opiskelijoille vaihtoehtoisena tapana visualisoida Java-ohjelmien suoritusta. Lisäksi Jeliotia on kehitetty ViSCoS-opiskelijoilta saadun palautteen perusteella. Visualisointityökalut eivät kuitenkaan ViSCoS-opinnoissa ole niin yleisesti käytössä, että niitä voitaisiin hyödyntää suoraan arvioinnissa.

3.4.5 Tietokoneavusteinen vertaisarviointi

Tietokoneavusteisella vertaisarvioinnilla Carter *et al.* (2003) tarkoittavat arviointimenetelmää, jossa vertaisarvioinnin järjestämiseksi hyödynnetään tietokonetta. Tämä voi tapahtua esimerkiksi arpomalla arvostelijat ja muuttamalla vastaukset sellaisiksi, etteivät arvostelijat tiedä kuka alkuperäinen vastaaja on.

ViSCoS-opinnoissa vertaisarviointia hyödynnetään nykyisin vain Tietojenkäsittelytieteen tutkimusaloja -kurssilla. Tällä hetkellä käytössä oleva menetelmä ei kuitenkaan ole tietokoneavusteinen. Moodle-ympäristön työkalut mahdollistavat tietokoneavusteisen vertaisarvioinnin järjestämisen käytännön toteutuksen. Tarkastelen vertaisarviointia tarkemmin luvussa 4.3.

3.5 Verkkotentit ja tunnistautuminen verkossa

Perinteisten tenttien suurin etu on se, että tenttitilaisuudessa tentin valvoja pystyy tarkastamaan tenttiin osallistuvien henkilöllisyyden. Tämän kaltainen tunnistaminen ei onnistu Internetin välityksellä. Todellisuudessa ei voida koskaan olla varmoja kuka tehtäviin todellisuudessa vastaa, jos tentti tai muu arviointi suoritetaan Internetin välityksellä. Tätä ongelmaa on pyritty kiertämään erilaisin tavoin.

Bari *et al.* (2004) esittelevät skenaarion, jossa verkossa opiskelevaa valvotaan *nettikameran* avulla. Nettikameraksi kutsutaan pientä videokuvaa Internetiin lähettävää kameraa, jotka ovat yleistyneet viime aikoina. Kirjoittajien suosittelemassa järjestelmässä oletetaan, että opiskelijan toimintaa voidaan seurata hänen oman nettikameransa välityksellä. Bari *et al.* mukaan järjestelmällä voidaan valvoa kuka päätteellä työskentelee.

Valitettavasti teknisesti taitava opiskelija pystyy helposti syöttämään videokuvana haluamaansa videomateriaalia. Video voi olla myös ennalta nauhoitettua. Tämä voidaan toki estää kysymällä opiskelijalta esimerkiksi ennalta arvaamaton kysymys. Opettaja ei kuitenkaan todellisuudessa voi tietää onko videolla esiintyvä vastauksia antava sama opiskelija, jonka opintosuoritusta ollaan arvostelemassa. Vaikka nettikamerat ovat nykyisin edullisia, kamera ei kuitenkaan välttämättä ole kaikkien saatavilla. Koska ViSCoS-opintoja tarjotaan suoritettavaksi myös ns. kehittyvissä maissa, nettikamera tai sellainen Internet-yhteys, jolla kuvaa voidaan välittää hyvätasoisena, on lähes kohtuuton vaatimus sillä saavutettuun hyötyyn nähden.

Eräs tapa yrittää estää vilppiä verkossa tapahtuvassa arvioinnissa on hyödyntää verkkotenttimiseen laadittuja ohjelmia. Cordova ja Thornhill (2007) esittelevät lyhyesti Lockdown Browser -nimisen ohjelmiston. Ohjelmisto rajoittaa tukemillaan selaimilla verkkosivujen selauksen vain sallituille sivuille. Lisäksi ohjelma estää muiden ohjelmien, kuin verkkoselaimen käyttämisen tentin suorituksen aikana. Tällaisen ohjelman hyöty on kuitenkin kyseenalainen. Nokkela opiskelija kiertää ohjelmiston rajoitukset kohtuullisen helposti. Ohjelmisto sitoo myös käyttäjän käyttämään tiettyä käyttöjärjestelmää ja WWW-selainta.

ViSCoS-kurssit on alusta lähtien laadittu niin, että opiskelija voi suorittaa opinnot millä tahansa käyttöjärjestelmällä ja valitsemallaan WWW-selaimella. Eräs tärkeimmistä syistä siihen, miksi Java-kieli valittiin ViSCoS-opinnoissa opetuskieleksi, oli se, että Java on käyttöjärjestelmäriippumaton ohjelmointikieli. Yksittäisen käyttöjärjestel-

män tai selaimen vaatiminen opiskelijoilta ei sovi ViSCoS-opintojen pohjana olevaan CANDLE-malliin. Järjestelmä, jonka Cordova ja Thornhill (2007) esittelevät, ei myöskään mitenkään takaa sitä, että verkkotentin suorittaa juuri se opiskelija, jonka opintosuoritusta ollaan arvostelemassa.

Myös monia muita tapoja verkkotunnistautumiseen ja etätenttien järjestämiseen on esitelty. Valitettavasti yksikään tapa ei tuo vastausta perusongelmaan. Koskaan ei voida olla varmoja siitä, kuka arvioitavan työn todellisuudessa laatii. Thomas *et al.* (2001) edottavat, että tentin jälkeen opettaja voisi soittaa puhelimitse muutamille opiskelijoille ja kysyä tarkentavia kysymyksiä tenttiin liittyen. Kirjoittajien mukaan tämä toimisi opiskelijoille eräänlaisena pelotteena. Valitettavasti puhelimenkaan välityksellä ei voida olla varmoja siitä, kuka puhelimesta todellisuudessa puhuu. ViSCoS-opintojen Perusopintojen harjoitustyö -kurssilla on kuitenkin käytetty opiskelijoille pakollista puhelinohjausta sekä opiskelijapalautteen keräämiseen että tapana vähentää vilpin tekoa kurssilla.

Vaikka verkkotenttien yhteydessä hyödynnettäisiin biometrista tunnistautumista, jossa opiskelija todentaa osallistumisensa arviointiin esimerkiksi sormenjäljen avulla, voi todellinen työn laatija olla joku toinen henkilö. Riittää että se joka tenttiä on tekevinään on paikalla esittäen sormenjälkensä. Toinen henkilö voi sitten hänen sijassaan vastata varsinaisiin tenttikysymyksiin. Simon (2005) kysyykin artikkelissaan, onko ylipäätään mitään järkeä laatia verkkotenttiä vilpin esiintymisen suuren todennäköisyyden takia.

Simon (2005) asettaa samalla kyseenalaiseksi koko verkko-opetuksen mielekkyyden. Samalla tavoin on epäselvää laatiiko opiskelija vastaukset myöskään harjoitustehtäviin kurssin aikana. Kirjoittaja kuitenkin toteaa, että näin kyyninen lähestymistapa ei nykyisin voi toimia, sillä verkko-opinnoille on olemassa suuri kysyntä. Simon (2005) pitääkin vilpin tunnistamista ja kitkemistä tärkeänä, mutta katsoo, että vilpin tekemistä ei voida koskaan täysin aukottomasti estää. Tämä kuitenkin koskee sekä verkko-opintoja että lähiopetusta.

Jos halutaan olla aivan varmoja, että opiskelija tekee tenttinsä itse, tentti on järjestettävä valvotuissa olosuhteissa. Thomas *et al.* (1998) ovat päätyneet järjestämään verkkotentin yliopiston tietokonealuokassa. Tämä mahdollistaa sen, että opiskelijoiden on tunnistauduttava. Myös tällaisessa tentin järjestämisessä on kuitenkin omat ongelmansa. Jos halutaan olla varmoja siitä, että opiskelijat eivät keskustele keskenään pikaviestimillä

tai sähköpostitse, täytyy tentin valvojan pystyä tehokkaasti seuraamaan opiskelijoiden työskentelyä.

Käytännössä rajoitetun ja valvotun työskentely-ympäristön järjestäminen luokkatilassa on mahdollista. Tällainen rajoitettu — ja rampautettu — ympäristö ei kuitenkaan vastaa todellista työympäristöä. Voidaankin kysyä poikkeako tällaisessa tilanteessa suoritettu tentti yhtään perinteisestä tentistä. Keinotekoiset rajoitukset opiskelijoiden viestimiselle lisäävät tilanteen kaksoisrakenteisuutta.

Kaiken kaikkiaan verkkotenttimiseen ja muihin Internetissä järjestettäviin arviointeihin liittyy kohtuullisen suuri riski vilpilliseen toimintaan. Viime kädessä on kuitenkin kysymys siitä, miten paljon opiskelijoihin voidaan luottaa. Opiskelijan valvontaan voidaan käyttää ja kehittää erilaisia menetelmiä, mutta mikään näistä menetelmistä ei ole aukoton tai sellainen, ettei sitä voisi teknisesti taitava opiskelija kiertää.

4 Yleisesti käytettyjen arviointimenetelmien esittely

Tässä luvussa erittelen tarkemmin erilaisia arviointimenetelmiä. Arviointimenetelmiä on kehitetty valtava määrä ja lisää julkaistaan jatkuvasti. Otan mukaan tarkasteluun yleisimmin käytettyjä verkko-opintoihin soveltuvia arviointimenetelmiä, sekä sellaisia menetelmiä, joita on aikaisemmin hyödynnetty tietojenkäsittelytieteen verkko-opinnoissa.

Esitellessäni arviointimenetelmät, kuvaan lyhyesti sitä, miten kutakin arviointimenetelmää käytetään ja miten arviointimenetelmä toimii. Nostan esille menetelmien vahvuuksia ja heikkouksia, sekä sitä millaisille kursseille ne soveltuvat. Jokaisesta arviointimenetelmästä on olemassa myös useita muunnoksia ja ideaalitapauksessa arviointimenetelmä räätälöidään vielä uudelleen kurssikohtaisesti (Karjalainen ja Kemppainen, 1994).

Luvussa 4.1 esittelen perinteisen tentin ja sen johdannaiset. Erilaisiin etätenttimenetelmiin ja niiden käyttämiseen keskityn luvussa 4.2. Muita tietojenkäsittelytieteen verkkokurssilla käyttökelpoisia menetelmiä käsittelen luvussa 4.3. Viimeiseksi luvussa 4.4 käyn läpi usean menetelmän samanaikaista käyttämistä arviointiin.

4.1 Perinteinen tentti ja sen johdannaiset

Perinteistä kirjallista tenttiä nimitetään myös valvotuksi kirjallinen tentiksi. Perinteistä tenttiä tarkastelin jo aiemmin luvussa 2.1. Kirjallisessa tentissä opiskelija vastaa opettajan laatimiin kysymyksiin kuulustelutilaisuudessa paikalla olevan valvojan läsnä ollessa. Paperi tai paperit, joille opiskelija vastauksensa tuottaa toimitetaan tentin tarkastavalle opettajalle. Valvoja valvoo, etteivät tenttiin osallistuvat tee mitään vilpillistä.

Perinteisen tentin ehdoton etu on se, että tentin valvoja voi tarkastaa tenttiin osallistuvien henkilöllisyyden. Näin voidaan varmistua, että tenttiin vastaaja on todellakin itse opiskelija. Perinteisen tentin ongelmana on siihen liittyvä kaksoisrakenteisuus. Opiskelijalla ei ole yleensä yliopistossa mahdollisuutta tutustua tenttinsä arvosteluun, joten arviointi ei ole kovinkaan avointa, vaikka arviointiperusteet olisivatkin julkisia.

*Materiaalitentti*⁵, jota nimitetään myös aineistotentiksi on kuten valvottu kirjallinen tentti, mutta opiskelija saa tuoda kuulustelutilaisuuteen mukanaan esimerkiksi oppikirjan tai jotain muuta opettajan sallimaa materiaalia (Karjalainen ja Kempainen, 1994). Materiaalitentti vastaa toteutukseltaan ja ongelmiltaan hyvin paljon perinteistä tenttiä. Opettajan kannalta materiaalitentin laatiminen voi olla haastavampaa kuin perinteisen tentin tekeminen.

Koska opiskelijoilla on mahdollisuus tuoda materiaalitenttiin mukanaan lähdemateriaalia, syntyy helposti kiusaus tehdä tentistä tarpeettoman hankala. Aineistotentti saattaa vaatia myös opiskelijoilta totuttelua. Hyppönen (2004) pitää mahdollisena myös, että opettaja tulee kysyneeksi helposti materiaalitentissä jotain sellaista, jota kurssin aikana ei ole harjoiteltu.

Materiaalitentin ajatuksena on ohjata oppimista perinteisen tentin kaltaisesta ulkoa opettelusta olemassa olevan tiedon hyödyntämiseen. Tenttimenetelmä soveltuu käytettäväksi myös ViSCoS-ympäristöön, mutta valitettavasti suurin osa ViSCoS-oppimateriaalista on Internetissä julkaistuja verkkosivuja. Tämän oppimateriaalin hyödyntäminen aineistotentissä on hyvin kankeaa, sillä sivustoja ei ole suunniteltu käytettäväksi tulostettuna.

Eräs aineistotentin versio on antaa opiskelijan laatia itse tenttiin mukaan otettava materiaali esimerkiksi käsin kirjoittamalla A4-arkeille. Daly ja Waldron (2004) jopa antavat ymmärtää, että tällaisen sallitun *lunttilapun* laatiminen voi olla eduksi oppimiselle ohjelmointikurssilla. Valitettavasta he eivät ole tehneet tukimusta aiheesta.

Suullisella tentillä tarkoitetaan kuulustelua, jossa opettaja henkilökohtaisesti kuulustelee opiskelijalta kurssiin liittyviä kysymyksiä ja opiskelija vastaa näihin kysymyksiin suullisesti (Gharibyan, 2005). Opettaja voi tehdä keskustelun aikana tarkentavia kysymyksiä tai johdatella opiskelijaa esittämään käsiteltävä asia ymmärrettävästi.

Teknisesti suullinen kuulustelu onnistuu myös etäopetuksessa, mutta valitettavasti opettajan on mahdotonta seurata opiskelijan nonverbaalista viestintää kun kuulustelu tehdään esimerkiksi puhelimen tai ip-puhelimen välityksellä. Tällaisen tekniikan avulla ei myöskään voida varmistua opiskelijan henkilöllisyydestä. Suullisen tentin ongelmana on se, että varsinainen tenttitilanne voi opiskelijan kannalta olla ahdistava (Hyppönen, 2004).

⁵Englanniksi open book exam

Mikäli opettaja ei osaa laukaista opiskelijan jännitystä, opiskelijan voi olla mahdotonta vastata esitettyihin kysymyksiin oman taitotasonsa mukaisesti. Suullinen tentti on myös opiskelijan kannalta melko luonnoton tilanne. Tenttiin on vaikea valmistautua ja on vaarana, että tenttiin valmistautuminen muistuttaa samanlaista ulkoa opiskelua kuin perinteiseen tenttiin valmistautuminen. Karjalainen ja Kemppainen (1994) ohjeistavat, että mikäli suullisia tenttejä käyttää opetuksessa, tenttien tulisi olla ennemminkin opetuskeskusteluja kuin kuulusteluja.

Monivalintakokeita käytetään yleisesti arviointiin erityisesti Pohjois-Amerikassa. Monivalintakokeissa on useita kysymyksiä, jossa jokaisessa on useampia vastausvaihtoehtoja (Karjalainen ja Kemppainen, 1994). Opiskelija valitsee näistä vastausvaihtoehtojista mielestään oikean vastauksen. Kuulustelu, jossa käytetään monivalintakokeita, voidaan helposti järjestää myös erilaisissa verkko-oppimisympäristöissä.

Hyvän monivalintakokeen laatiminen on usein vaikeaa (Karjalainen ja Kemppainen, 1994). On kohtuullisen helppoa laatia monivalintakoe, jossa mitataan ulkoa muistamista. Tiedon soveltamista on paljon hankalampaa arvioida monivalintakokeiden avulla. Woodford ja Bancroft (2005) kuitenkin osoittavat, että monivalintakokeita voidaan tehokkaasti hyödyntää myös tietojenkäsittelytieteen opetuksessa, kunhan kysymykset laaditaan huolellisesti.

4.2 Etätentit

Kotitentti vastaa materiaalitenttiä, mutta opiskelija vastaa kysymyksiin omalla ajallaan tiettyyn määräaikaan mennessä (Karjalainen ja Kemppainen, 1994). Kotitenttin suorituspaikkaa ei ole rajattu. Kotitentti ei rajoitu pelkästään perinteisiin tenttimuotoihin, vaan myös essee voidaan lukea kuuluvaksi kotitentiksi. Opettajalla ei ole mahdollisuutta kotitentissä valvoa sitä, kysyykö opiskelija joltain toiselta neuvoa kuulusteluun vastaamiseksi tai vastaako joku toinen opiskelijan puolesta tenttikysymyksiin.

Tenttitilaisuudet aiheuttavat usein jännitystä ja jopa ahdistusta⁶. Kotitentit mahdollistavat tenttiin vastaamisen tutussa ympäristössä vähentäen kuulusteluun liittyviä paineita. Tästä johtuen kotitentteihin liittyy kuitenkin suuri vilpin esiintymisen riski. Koska opettaja ei voi valvoa opiskelijoiden työskentelyä kotitenttin aikana jää arvioitavan oma panos kotitentissä opettajan ja opiskelijan välisen luottamuksen varaan.

⁶Katso esimerkiksi Committee on the Foundations of Assessment, National Research Council (2001)

Verkkotentissä opiskelija vastaa tietoverkossa eli nykyisin useimmiten Internetissä oleviin tenttikysymyksiin (Hyppönen, 2004). Tenttiin vastaamisen on usein tapahduttava tietyn ajan kuluessa sen aloittamisesta. Tentissä käytetään tietokonetta ja vastaukset lähetetään opettajalle tietoverkon välityksellä. Verkkotenttejä voidaan järjestää joko WWW-selaimella tai erikseen etätenttimistä varten laaditun ohjelmiston avulla. Verkkotenttien etuna on, että opiskelijat voivat suorittaa tentin mistä tahansa, eikä heidän tarvitse erikseen tentin takia matkustaa erilliseen tenttitilaisuuteen. Verkkotentteillä voidaan myös vähentää tenttiin liittyvää jännitystä, sillä tentti on mahdollista tehdä tutussa ympäristössä.

Verkkotenttien ongelmat vastaavat kotitenttien ongelmia. Koska tenttiä ei voida valvoa perinteisin menetelmin, ei voida olla varmoja, että opiskelija vastaa tenttiin annettujen ohjeiden mukaan itsenäisesti. Myöskään tenttiin vastaajan henkilöllisyydestä ei voida varmistua. Vilpin estämiseen verkkotentteissä on kehitetty erilaisia menetelmiä. Mikään nykyisin käytössä olevista menetelmistä ei kuitenkaan varmista sitä, että vilppiä ei tapahdu, kuten luvussa 3.5 todetaan.

4.3 Muut arviointimenetelmät

Essee on lyhyehkö vapaamuotoinen yleistajuinen kirjoitelma, jossa opiskelija pyrkii analysoimaan ja erittelemään aiheena olevaa kysymystä tai ongelmaa lähdekirjallisuuden avulla tuoden esille myös omat mielipiteensä (Karjalainen ja Kempainen, 1994). Opettajalla ei koskaan ole täyttä varmuutta siitä onko opiskelija laatinut esseeseen itse. Esseet ovat opettajan näkökulmasta joskus hitaampia ja hankalampia arvostella, kuin perinteiset tentit. Myös yhdenmukaisia arvostelukriteereitä on vaikea laatia, sillä opettajan arvio esseestä perustuu hänen subjektiiviseen mielipiteeseensä työn tasosta.

Jotta arviointi on läpinäkyvää opiskelijalle, hänen tulisi saada työstään perusteltu kirjallinen arviointi. Essee soveltuu erityisen hyvin kursseille, joissa opiskelijan halutaan pohtivan ja analysoivan kurssilla käsiteltyjä asioita. Automaattisen esseiden tarkastamisen menetelmiä ja siihen liittyviä ongelmia käsiteltiin aikaisemmin luvussa 3.4.2.

Harjoitustyö on oppimisen arviointimenetelmä, jossa opiskelija keskittyy itsenäisesti harjoittelemaan jotain opiskeltavaa asiaa syvällisemmin (Karjalainen ja Kempainen, 1994). Harjoitustyö voidaan arvioida lopputuloksen, työprosessin tai näiden molempien perusteella. Harjoitustyön tarkempi rakenne riippuu siitä, mitä opiskelija on

opiskelemissa tai harjoittelemassa. Ohjelmoinnissa harjoitustyö voi olla esimerkiksi ohjelman laatiminen ja laaditun ohjelman dokumentointi. Harjoitustyö voi myös olla esimerkiksi pienimuotoinen tieteellistä tutkimusta muistuttava kirjoitelma.

Harjoitustyön arviointi perustuu usein opettajan subjektiiviseen arviointiin. Jotta harjoitustyön arviointi olisi mahdollisimman avointa opiskelijan kannalta, opiskelijoiden tulisi tuntea arviointikriteerit sekä saada työstään kirjallinen palaute. Harjoitustöiden aiheiden laatiminen ja harjoitustyön ohjaaminen vaatii usein paljon opettajan resursseja (Hyppönen, 2004).

Oppimispäiväkirjalla tarkoitetaan opiskelijan laatimaa päiväkirjatyypistä esitystä omasta oppimisprosessistaan kurssin aikana (Hyppönen, 2004). Oppimispäiväkirjan kirjoittaminen vaatii opiskelijalta paneutumista ja aluksi oman oppimisprosessin kuvaaminen päiväkirjamuodossa voi olla tällaiseen työskentelytapaan tottumattomille opiskelijoille hankalaa.

Oppimispäiväkirja kertoo opettajalle opiskelijan oppimisprosessista ja tuo ilmi millaisia ongelmia opiskelijalla on opintojakson aikana ollut. Oppimispäiväkirjojen arviointi on kohtuullisen työlästä. Myös oppimispäiväkirjojen arvioinnin tulee olla opiskelijoille läpinäkyvää, jotta opiskelija näkee mitä asioita opettaja painottaa oppimispäiväkirjaa arvostellessaan. Karjalainen ja Kemppainen (1994) suosittelevat, että arvioitu oppimispäiväkirja tulisi palauttaa opiskelijalle etenkin jos se on käsin kirjoitettu.

Oppimispäiväkirjoja voidaan käyttää hyvin erilaisilla kursseilla joko osana arviointia tai koko kurssin arvioinnin välineenä. George (2002) hyödynsi oppimispäiväkirjoja ohjelmointikurssilla ja sai oppimispäiväkirjan pitämisestä kurssilla hyviä tuloksia. Kirjoittajan mukaan pääosa opiskelijoista piti oppimispäiväkirjaa hyödyllisenä oman oppimisen kannalta. Osan opiskelijoista oli kuitenkin vaikea yhdistää oppimispäiväkirjan kirjoittamista ohjelmoinnin opiskeluun. Oppimispäiväkirjalla voidaan kirjoittajan mukaan valmentaa opiskelijoita myös varsinaiseen ohjelmistotuotantoprosessiin, jossa oman työn arviointi ja seuraaminen on tärkeitä.

Portfolio eli työkansio on kokoelma opiskelijan töitä, joilla opiskelija osoittaa osaamisensa ja edistymisensä (Karjalainen ja Kemppainen, 1994). Portfolion ajatuksena on, että opiskelija kokoaa osan edustavimpia töitä näytteeksi osaamisestaan. Portfolioon voidaan liittää myös oppimispäiväkirja tai jokin muu opiskelijan laatima dokumentti opiskelun etenemisestä. Portfolioiden käyttö on yleistä erityisesti taideaineissa. Portfo-

lioita on kuitenkin käytetty menestyksellisesti myös tietojenkäsittelytieteen opetuksessa (Ury ja McFarland, 2001). Monet verkko-oppimisympäristöt tarjoavat työvälineitä portfolioiden laatimiseksi. Portfolion laatimisen ohjeistaminen voi olla hankalaa.

Miellekartta (Mind Map) on puumainen graafinen esitys jostakin asiasta ja siihen liittyvistä asioista (Hyppönen, 2004). Miellekarttaa voidaan käyttää muiden arviointivälineiden osana tai arviointi voidaan suorittaa kokonaan miellekartan avulla. Miellekartan ajatukseen liittyy, että karttaan kuvataan sillä analysoitava käsite subjektiivisesti. Tämä tekee miellekartan arvioimisen haasteelliseksi, sillä toisen laatimaa miellekarttaa voi olla vaikea ymmärtää.

Miellekartan kaltainen on sitä läheisesti muistuttava *käsitekartta*. Käsitekartta on Meisalo *et al.* (2003) mukaan rakenteeltaan tarkemmin rajattu kuin miellekartta. Käsitekartassa käsitteiden väliset suhteet nimetään ja kuvataan tarkasti. Käsitekarttaa käytetään tiedon strukturoidussa esittämisessä eikä se salli samanlaista vapaa assosiaatiota kuin miellekartta.

Ryhmätyö vastaa harjoitustyötä, mutta joukko opiskelijoita tuottaa ryhmätyön yhdessä (Karjalainen ja Kemppainen, 1994). Yleensä ryhmätyö voi olla laajempi kuin harjoitustyö, sillä työtä on tekemässä useampia opiskelijoita. Ryhmätyötä arvioidessa erityisen vaikeaksi opettajan kannalta muodostuu sen arviointi, ovatko kaikki ryhmän jäsenet antaneet yhtä suuren panoksen työn tekemiseen. Erilaisilla opiskelijaryhmillä on myös täysin erilaiset motivaatiot ja työskentelytavat opiskelussaan. Vaikka verkko-oppimisympäristöissä on erilaisia työkaluja ryhmätöiden tekemisen tueksi, ryhmätyö verkko-opinnoissa on haastavampaa kuin tilanteessa, jossa opiskelijat voivat työstää työtä samassa paikassa. Ryhmätyöllä todettu olevan positiivisia vaikutuksia opiskelijoiden oppimiseen tietojenkäsittelytieteen kurssilla (Melin ja Cronholm, 2004).

Vertaisarviointi, jota nimitetään myös toveriarvioinniksi on arviointimenetelmä, jossa ainakin osan arvioinnista suorittavat muut opiskelijat (Karjalainen ja Kemppainen, 1994). Vertaisarviointia voidaan käyttää minkä tahansa suorituksen tai työn arviointiin. Vertaisarvioinnin suurimpana etuna on, että osa arviointivastuusta siirtyy opiskelijoille ja varsinainen arviointi otetaan mukaan osaksi oppimisprosessia. Sitthiworachart ja Joy (2004) esittelevät sovelluksen, jonka avulla vertaisarviointia on hyödynnetty menestyksellisesti ohjelmointitehtävien arvioinnissa. Suurimpana haasteena vertaisarvioinnissa on sen käytännön organisointi, sekä se miten luotettavia yksittäiset vertaisarviot

ovat (Hyppönen, 2004). Karjalainen ja Kemppainen (1994) pitävät vertaisarvioinnin hallitsemista tärkeänä työelämään siirryttäessä.

Itsearviointi on arviointimenetelmä, jossa opiskelija arvioi itse omaa osaamistaan (Hyppönen, 2004). Karjalainen ja Kemppainen (1994) pitävät itsearviointin oppimista erittäin tärkeänä, sillä opiskeluajan jälkeen itsearviointitaitoja tarvitaan työyhteisössä. Itsearviointi voi myös lisätä opiskelijan vastuullisuutta omassa opiskelussaan. Itsearviointia voidaan hyödyntää osana arviointia tai arviointi voi perustua kokonaan opiskelijan omaan arviointiin. Suurin itsearviointin hankaluus on se, että kukin opiskelija arvioi omaa osaamistaan omista näkökohdistaan. Tällöin arviointikriteerit eivät välttämättä ole yhtäläiset.

Dialogitentit ovat yliopisto-opintojen arviointiin tarkoitettuja arviointimenetelmiä, jotka on kehitetty vastamaan luvussa 2.6 listattujen hyvän arviointimenetelmän ominaisuuksia. Dialogitenteissä opiskelijat laativat kurssin keskeisistä ja epäselviksi jääneistä asioista lyhyen esseen kaltaisen kirjallisen tuotoksen. Arvioinnin seuraavassa vaiheessa käydään opetuskeskustelu opettajan ja opiskelijan välillä. Keskustelun aikana analysoidaan, järjestellään, selvennetään ja arvioidaan kurssin sisältöä opiskelijan kirjallisen työn pohjalta (Karjalainen ja Kemppainen, 1994).

Karjalainen ja Kemppainen (1994) esittelevät dialogitenteistä myös ryhmäversiot eri kokoisille ryhmille, sekä laajempien kokonaisuuksien arviointiin tarkoitetun dialogitenttien variaation. Kirjoittajien mukaan dialogitentti on suunniteltu yliopiston tiedontuottamisluonteen pohjalta. Dialogitenttien ajatuksena on eritellä ja analysoida arvioitavaa asiasisältöä niin, että arviointitilaisuudesta hyötyvät niin opiskelija kuin opettajakin. Dialogitentissä itse opiskeltava asia nostetaan tärkeimmäksi. Mikäli arvioinnissa tähdätään arvosanan antamiseen, se päätetään keskustelutilaisuudessa opettajan ja opiskelijan yhteisellä päätöksellä.

Dialogitenttien hyödyntäminen verkko-opinnoissa on kohtuullisen haastavaa. Puhelimen tai ip-puhelimen välityksellä kommunikointi ei vastaa tilannetta, jossa sekä opiskelija että opettaja ovat samassa fyysisessä tilassa. Sähköpostilla tai pikaviestimessä käytävä keskustelu ei myöskään korvaa aitoa kontaktiopetuksen vuorovaikutustilannetta. Toimivan dialogitenttien räätälöiminen esimerkiksi ohjelmointikurssille voi olla hyvin hankalaa opetettavan asian luonteen vuoksi.

Matriisiarviointi eli rubriikkiarviointi on arviointimenetelmä, jossa arviointiperusteet taulukoidaan ymmärrettävään muotoon ja erilaisten taitotasojen vaatimukset esitetään ymmärrettävästi (Karjalainen, 2001). Matriisiarviointi soveltuu kaikenlaiseen arviointiin. Matriisin käyttäminen selventää arviointiperusteita sekä opettajille, että opiskelijoille. Kun arviointiperusteet ja oppimistavoitteet on selkeästi kirjattu, arvostelusta tulee avoimempaa ja läpinäkyvämpää. Samalla matriisiarvioinnin käyttäminen auttaa opiskelijoita sitoutumaan opetukseen, koska oppimisen tavoitetasot on määritelty etukäteen. Karjalainen (2001) suosittelee matriisin laatimista yhteistyössä opiskelijoiden kanssa. Esimerkki arviointimatriisin laatimisesta on esitetty liitteessä 1. Moskal *et al.* (2002) raportoivat arviointimatriisien olevan hyödyllisiä arvosteltaessa tietotekniikan etiikkaan liittyviä esseitä.

4.4 Arvioinnin hajauttaminen

Kurssin tai opintosuorituksen arviointia ei ole välttämätöntä sitoa ainoastaan yhteen arviointimenetelmään, vaan arvioinnissa voidaan käyttää useampia arviointimenetelmiä. Tällöin kurssin arvioinnissa käytetään kurssin aikana kerättyä arviointitietoa ja painotetaan lopullinen arvostelu, sen mukaan mikä katsotaan kurssin lopullisen arvioinnin kannalta tarkoituksenmukaiseksi. Varsinkin laajemmilla kursseilla arvioinnin hajauttaminen useampaan osaan on usein tarkoituksenmukaista. Mitä enemmän erikseen arviointia vaativia osia kurssilla sitä suuremmaksi kasvaa arviointiin tarvittava työmäärä.

Yksittäinen kurssi on myös mahdollista arvioida niin, että arviointiprosessi kestää koko kurssin ajan. Tällöin esimerkiksi kurssin aikana tehtyjen yksittäisten harjoitustehtävien arviointi määrittää opiskelijan kurssin kokonaisarvioinnin tuloksen. Tällaista arviointimenetelmää kutsutaan *jatkuvaksi arvioinniksi* (Karjalainen ja Kemppainen, 1994). Jatkuvan arvioinnin etuna on, että opiskelijoiden yksittäisten suoritusten stressaavuus vähenee. Verkko-opetuksessa jatkuvan arvioinnin ongelmat vastaavat yksittäisten harjoitustehtävien palauttamiseen liittyviä ongelmia. Koska opettajalla ei ole mahdollisuutta seurata opiskelijan työskentelyä, vilpin esiintyminen on mahdollista.

5 ViSCoS-opinnot

Syksyllä 2000 Joensuun yliopiston tietojenkäsittelytieteen laitos tarjosi ensimmäistä kertaa Pohjois-Karjalan lukiolaisille Internetin välityksellä 15 opintoviikon laajuisen *virtuaaliapprobatur*-opintokokonaisuuden (Haataja *et al.*, 2001). Myöhemmin virtuaaliapprobatur-opinnot nimettiin uudelleen *ViSCoS*-opinnoiksi (Virtual Studies on Computer Science). Nimi vaihdettiin, koska approbatur-termi poistui tutkintorakennemuutoksen myötä käytöstä ja opintokokonaisuuden nimestä haluttiin kansainvälisemmin vetoavampi.

Opintojen alkuperäisenä ajatuksena oli, että opintokokonaisuus tarjosi lukiolaisille mahdollisuuden opiskella Internetin välityksellä tietojenkäsittelytieteen 15 opintoviikon (nykyisin 25 opintopisteen) laajuiset approbatur- eli perusopinnot jo lukioaikana. ViSCoS-opinnot vähintään hyvin tiedoin suorittanut opiskelija sai opiskelupaikan Joensuun yliopiston tietojenkäsittelytieteen laitokselta.

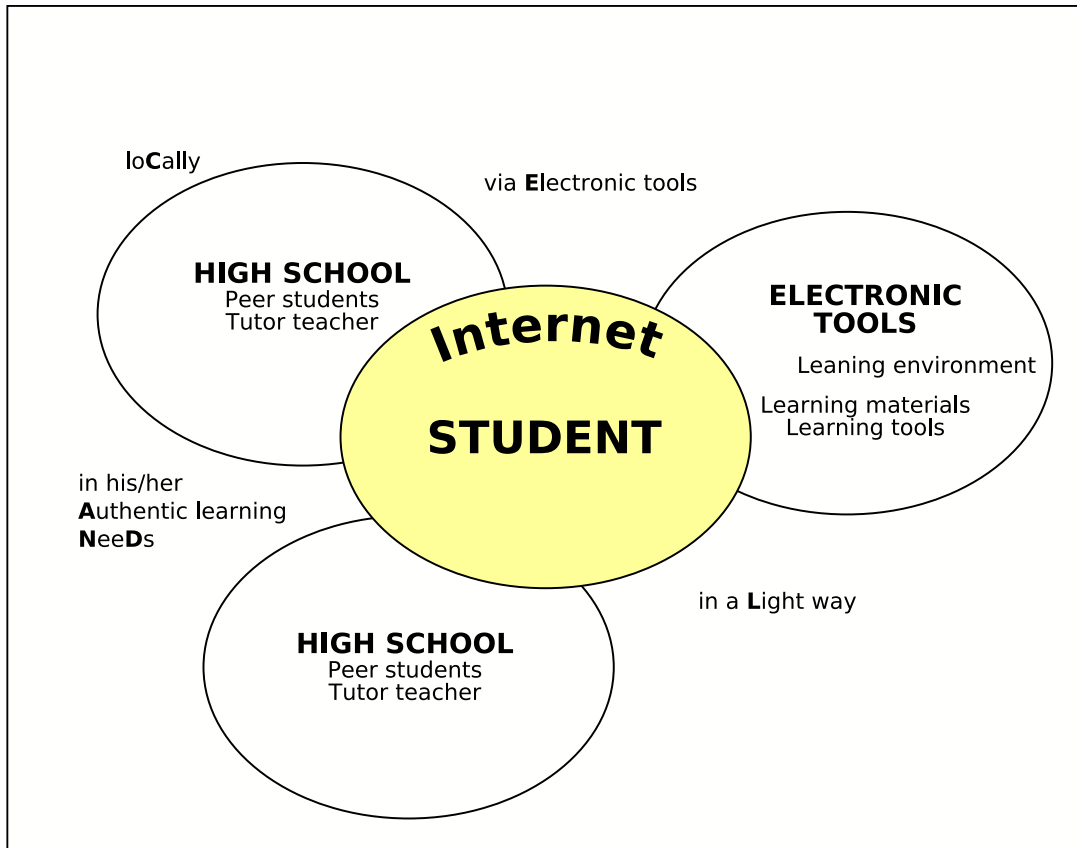
Esittelen tässä luvussa ViSCoS-opintojen rakenteen ja suoritustavat. Lisäksi käyn läpi sen, kuinka opiskelijoiden suoritukset arvioidaan tällä hetkellä ViSCoS-opinnoissa. Luvussa 5.1 esittelen ViSCoS-opintojen suunnitteluperiaatteet ja opintojen kehityksen ja luvussa 5.2 käyn läpi ViSCoS-kokonaisuuteen liittyvät kurssit sekä niiden suoritustavat. Arviointiin ViSCoS-opinnoissa keskityn tarkemmin luvussa 5.3.

5.1 ViSCoS-opintojen historia

Kun ViSCoS-opintoja alettiin suunnitella vuonna 2000, opintokokonaisuus rakennettiin siten, että lukio-opiskelijoiden oli mahdollista suorittaa opinnot puolentoista vuoden kuluessa. ViSCoS-opinnot kehitettiin alun perin ns. *CANDLE*-mallin perusteella (Sutinen ja Torvinen, 2003). *CANDLE*-malli on esitetty kuvassa 4.

CANDLE-mallin lähtökohtana oli aloittaa kehittämään tietojenkäsittelytieteen verkko-opintoja lukijoille ensi sijassa opiskelijalähtöisesti (Sutinen ja Torvinen, 2003). Tavoitteena oli saada opinnoista ajasta ja paikasta riippumattomat sekä tuottaa sellaista oppimateriaalia, joka toimi myös hieman vanhemmalla tietokonelaitteistolla ja hitaamalla Internet-yhteydellä.

Olellaisena osana CANDLE-malliin kuuluu myös opiskelijoiden henkilökohtainen ohjaus, joka otettiin tärkeäksi teemaksi jo ViSCoS-opintojen suunnittelun alkuvaiheessa (Sutinen ja Torvinen, 2003). Eräs CANDLE-mallin suunnitteluajatuksista oli tarjota linkki painetun, sekä viikoittaisten harjoitustehtävien ratkaisemiseksi tarkoitettun Internetissä julkaistun oppimateriaalin välillä.



Kuva 4: CANDLE-malli (Sutinen ja Torvinen, 2003)

Aluksi virtuaaliapprobatur-kokonaisuutta tarjottiin Pohjois-Karjalan lukiolaisille. Joensuun yliopiston tietojenkäsittelytieteen laitos sekä Itä-Suomen virtuaaliyliopisto rahoittivat opinnot, joten opiskeleminen oli lukiolaisille ilmaista. Verkko-oppimateriaali tuotettiin yhteistyönä Joensuun yliopiston, Kuopion yliopiston, sekä Lappeenrannan teknillisen yliopiston⁷ toimesta. Verkko-oppimateriaalin tuottamishanketta koordinoitiin Itä-suomen virtuaaliyliopiston kautta.

Syksyllä 2001 virtuaaliapprobatur-opintoja tarjottiin suoritettavaksi myös Etelä-Savon lukioissa. Seuraavana syksynä virtuaaliapprobaturia hyväksyttiin suorittamaan myös

⁷Aikaisemmin Lappeenrannan teknillinen korkeakoulu.

Joensuun yliopiston Savonlinnan opettajankoulutuslaitoksen sivuaineopiskelijat sekä Päivölän kansanopiston matematiikkalinjalaiset.

Vuonna 2004 opinnoista tuli kaikille avoimia ja niitä pääsi suorittamaan kuka tahansa, sillä ViSCoS-opinnot siirryttiin tarjoamaan Joensuun yliopiston täydennyskoulutuskeskuksen Avoimen yliopiston kautta. Samalla opinnot muuttuivat kaikille opiskelijoille maksullisiksi. Opintonsa vähintään hyvin tiedoin suorittaneet opiskelijat hyväksytään edelleen tietojenkäsittelytieteen pääaineopiskelijoiksi ns. *avoimen yliopiston väylän* kautta.

ViSCoS:n alkuperäisenä ajatuksena oli markkinoida tietojenkäsittelytiedettä lukiolaisille mahdollisena opiskelualana ja tarjota opiskeltavaksi alan perusopintokokonaisuus. Lisäksi opintonsa hyvin tiedoin suorittaneet saivat mahdollisuuden siirtyä pääaineopiskelijoiksi tietojenkäsittelytieteen laitokselle. Toisaalta opintokokonaisuus tarjosi jo alusta lähtien mahdollisuuksia Internet-pohjaisten tietojenkäsittelytieteen opintojen kehittämiseen, sekä muodosti mielenkiintoisen tutkimusympäristön (Haataja *et al.*, 2001).

Vuonna 2006 ViSCoS-opintoja alettiin tarjota englanniksi. Opintoja ryhdyttiin samalla markkinoimaan suoritettavaksi myös ulkomailta käsin. Koska opintoja tarjottiin myös niin sanottuihin kehittyviin maihin, CANDLE-mallin mukainen oppimateriaalin keveys oli edelleen erittäin ajankohtaista. Varsinkaan kehittyvissä maissa kiinteä Internet-yhteys ei ole läheskään kaikkien potentiaalisten opiskelijoiden saavutettavissa.

5.2 ViSCoS-kurssit

ViSCoS-opintokokonaisuutta on kehitetty koko sen olemassaolon ajan. Opintojen painopistealueet ovat edelleen kuitenkin pääosin samanlaisia, kuin opintoja alun perin suunniteltaessa. Internetin välityksellä opiskelijoille tarjottavaa oppimateriaalia sekä opetustapoja on sen sijaan parannettu jatkuvasti.

Opettajat ovat kehittäneet ja laajentaneen vuosittain kurssikohtaista materiaalia. Esimerkiksi tietojenkäsittelytieteen pääaineopiskelijoilla teetettyjä ViSCoS-opintoihin liittyviä yksittäisiä aiheita syventäviä harjoitustöinä tehtyjä lisämateriaaleja on lisät-

ty osaksi opintojen aikana tarjottavia verkko-oppimateriaaleja. Tämän lisäksi kursseja on uudistettu ja ajanmukaistettu vuosittain.

Vuonna 2007 ViSCoS-kursseja yhtenäistettiin vastaamaan Joensuun yliopiston Tietojenkäsittelytieteen ja tilastotieteen laitoksen uusien tutkintovaatimusten mukaisia tietojenkäsittelytieteen perusopiskelijoiden opintoja. Johdatus algoritmiikkaan -kurssi jäi pois ja sen tilalle tuli Diskreetit rakenteet -kurssi. Lisäksi kurssien pituuksia ja painoituksia muutettiin hieman.

ViSCoS-opinnot on suunniteltu niin että opiskelijan on mahdollista suorittaa ne puolesatoista vuodessa. Käytännössä tämä tarkoittaa sitä, että varsinaista opetusta annetaan kolmen lukukauden ajan. Vaikka kurssit ovat Internetin välityksellä opiskeltavia verkkokursseja, niiden kurssimuotoisessa suorittamisessa on aikarajat, sillä kurssien aikana opiskelijoille tarjotaan henkilökohtaista ohjausta (Haataja *et al.*, 2001). Aikarajojen on myös huomattu motivoivan opiskelijoita, sillä kurssit joilla ei ole aikarajaa suoritetaan yleensä vasta viimeisenä.

Vuonna 2007 ViSCoS-opinnot koostuivat taulukko 2:n mukaisista opinnoista. Taulukossa on kuvattu myös lyhyesti kurssilla käsiteltävät asiat. Koska opinnot on suunniteltu niin, että ne voidaan suorittaa puolesatoista vuodessa, taulukkoon on merkitty suositeltu kurssin suorittamisaika jos opiskelija etenee opinnoissaan normaalissa aikataulussa.

Opiskelijat suorittavat ensimmäisen syksyn aikana Ohjelmointi 1 sekä Tietotekniikan perusteet -kurssit. Tämän jälkeen kevätlukukauden aluksi opiskelijat suorittavat Ohjelmointi 2 sekä Tietojenkäsittelytieteen tutkimisaloja -kurssit. Loppukeväästä ja keuhällä suoritusvuorossa on pääosin itsenäisenä opiskeluna Perusopintojen harjoitustyö -kurssi. Toisen vuoden syksyllä opiskelijat aloittavat koko syksyn kestävän Diskreetit rakenteet -kurssin ja suorittavat sen kanssa rinnakkain Tietokoneen rakenne ja käyttöjärjestelmä -kurssia. Johdatus tietojenkäsittelytieteen etiikkaan kurssin opiskelijat voivat suorittaa missä vaiheessa opintoja tahansa.

Kurssien yleisin suoritusmuoto on viikoittaisten harjoitustehtävien tekeminen ja kurssin loppukokeen hyväksytyt suorittaminen. Tämä suoritustapa on käytössä Ohjelmointi 1, Tietotekniikan perusteet, Ohjelmointi 2, Diskreetit rakenteet sekä Tietokoneen rakenne ja käyttöjärjestelmä -kursseilla. Opiskelijoilla on käytettävissään Moodle-oppimisympäristössä oleva kurssikohtainen verkko-oppimateriaali, joka tukee

Taulukko 2: ViSCoS opintojen (25 op) rakenne lukuvuonna 2006–2007

Kurssi	Laajuus	Kurssilla käsiteltäviä asioita	Suoritus aika
Tietotekniikan perusteet	2 op	Johdatus tietotekniikkaan sekä tietoverkkojen ja perustyövälineiden käyttö	1. syksy
Ohjelmointi osa 1	3 op	Johdatus ohjelmointiin Java-kielellä, ohjelmoinnin perusrakenteet sekä johdatus applettien ja grafiikan ohjelmointiin	1. syksy
Ohjelmointi osa 2	4 op	Johdatus olio-ohjelmointiin Java-kielellä, olio-ohjelmoinnin perusteet, graafiset sovellukset sekä tarkempaa tutustumista Java-kieleen	1. kevät
Tietojenkäsittelytieteen tutkimusaloja	2 op	Tutustuminen tietojenkäsittelytieteen alalla tehtävään tutkimukseen	1. kevät
Perusopintojen harjoitustyö	4 op	Itsenäinen laajahko ohjelmointityö ja sen dokumentointi	1. kevät ja kesä
Diskreetit rakenteet	5 op	Tietojenkäsittelijälle hyödyllisten matemaattisten peruskäsitteiden ja -työkalujen oppiminen. Kursin aiheita ovat mm. propositiologiikka, joukko-oppi, matemaattinen induktio, todistusmenetelmät sekä todennäköisyyslaskennan perusteet	2. syksy
Tietokoneen rakenne ja käyttöjärjestelmä	3 op	Tietokoneen rakenne ja toiminta, tietokoneen fysikaaliset perusteet ja mikro-ohjelmointi	2. syksy
Johdatus tietojenkäsittelyn etiikkaan	2 op	Johdatus tietojenkäsittelytieteen alaan liittyviin eettisiin ongelmiin	ei rajattu

viikoittaisten harjoitustehtävien tekemistä. ViSCoS-kurssien opettajat tarkistavat viikoittain opiskelijoiden palauttamat harjoitustehtävät ja antavat niistä sanallisen palautteen. Kurssin päätyttyä opiskelijat osallistuvat Avoimen yliopiston järjestämässä tenttipaikassa opettajien laatimaan kirjalliseen tenttiin kurssilla opetetuista aiheista.

ViSCoS-ohjelmointikurssien kehittämisessä on käytetty lähtökohtana ns. *visuals first* -lähestymistapaa (Kareinen *et al.*, 2002). Lähestymistavan ajatuksena on motivoida opiskelijaa Javan grafiikkaominaisuuksien perusteella. Suuri osa kurssien harjoitustehtävistä ja esimerkeistä liittyy grafiikan ohjelmointiin Java-kielellä. Esimerkkeinä ja harjoitustehtävinä on usein myös yksinkertaisia pikkupelejä. Kun ViSCoS-opintoja alun perin suunniteltiin, kohderyhmänä olivat pääosin lukioikäiset. Heille pelit ja pelien laatiminen oli luonteva keino tutustua ohjelmointiin. Ohjelmoidessaan omia pelejä opiskelijat pääsevät vapaasti käyttämään mielikuvitustaan, koska tehtäviä ei ole kovin tarkasti rajattu.

Johdatus tietojenkäsittelytieteen etiikkaan -kurssin opiskelijat suorittavat kirjoittamalla esseetyyppisen kirjoitelman jostakin tietotekniikkaan liittyvästä eettisestä ongelmasta. Tietojenkäsittelytieteen tutkimusaloja -kurssilla opiskelijat laativat useimmiten ryhmätyönä pienimuotoisen tutkimustyyppisen kirjoitelman jostakin tietojenkäsittelytieteen tutkimusalueesta. Perusopintojen harjoitustyö -kurssilla opiskelijat hyödyntävät oppimiaan ohjelmointitaitoja laatimalla ja dokumentoimalla itse laajemman Java-ohjelman.

5.3 Opiskelijan osaamisen arviointi nykyisissä ViSCoS-opinnoissa

Useimmilla ViSCoS-kursseilla käytetty arviointimenetelmä on perinteinen valvotuisissa olosuhteissa suoritettava kirjallinen tentti. Kurssin loputtua Avoin yliopisto järjestää opiskelijoille tenttipaikan, jossa opiskelijat vastaavat kirjallisesti opettajan etukäteen laatimiin kysymyksiin. Tenttitilaisuudessa on valvoja, joka tarkistaa opiskelijoiden henkilöllisyyden, sekä valvoo ettei vilppiä tapahdu.

ViSCoS-opinnoissa arviointi tähtää pääosin arvosteluun opiskelijoiden arvosanojen määrittämiseksi. Kaikki muut, paitsi johdatus tietojenkäsittelytieteen etiikkaan -kurssi arvostellaan numeroarvosanoin skaalalla 1-5. Käsittelen kursseilla nykyisin olevat arviointimenetelmät tarkemmin luvussa 6.2.

Kursseilla, joissa opiskelijat palauttavat viikoittaisia harjoitustehtäviä, opiskelijan tulee tehdä hyväksytysti vähintään kolmasosa kurssin harjoitustehtävistä, jotta hän voi suorittaa kurssin kurssitentillä. Mitä enemmän opiskelija tekee kurssin aikana harjoitustehtäviä, sitä enemmän hän saa *bonuspisteitä* kurssitenttiin. Bonuspisteet ovat ylimääräisiä pisteitä, jotka lasketaan kurssitentistä saadun pistemäärään ennen kurssiarvosanan laskemista. Ohjelmointikursseilla harjoitustehtävistä saa hieman enemmän bonuspisteitä tenttiin, sillä tehtävistä tekemisen on havaittu lisäävän opiskelijoiden motivaatiota harjoitustehtävien tekemiseen. Kurssit on mahdollista suorittaa myös yleisillä tenteillä.

Suurimpana haasteena ViSCoS-kurssien arvioinnissa on se, että ihannetilanteessa kurssit opetettaisiin ja arvioitaisiin täysin Internetin välityksellä. Tällä hetkellä kurssit kuitenkin arvioidaan kolmea kurssia lukuun ottamatta kurssin päätteeksi perinteisellä tentillä, jossa opiskelijan on oltava fyysisesti läsnä. Tenttien ja harjoitustöiden tarkastaminen vie kohtuullisen suuren osan opettajien työajasta. Erityisen ongelmallisia ovat opiskelijat, jotka suorittavat ViSCoS-opintoja ulkomailta käsin, sillä tenttien postitusaajat venyvät valitettavan pitkiksi.

Varsinaiseen arvosteluun johtavan arvioinnin lisäksi ViSCoS-kursseilla on viime vuosina alettu säännöllisesti kerätä opiskelijoilta viikoittaista itsearviointipalautetta ohjelmointikursseilla. Opettajat ovat kokeneet opiskelijoiden itsearvioinnin hyväksi tavaksi kehittää oppimateriaalia sekä tehtäviä. Lisäksi opiskelijat esittävät itsearvioinnin osana usein tarkentavia kysymyksiä opiskeltavaan asiaan liittyen. Opettajilla on itsearvioinnin kautta myös mahdollisuus seurata sitä, miten opiskelijat kokevat oppineensa käsillä olevan asian.

Opiskelijoita on myös kannustettu verkkokeskusteluun ja kysymysten esittämiseen kurssien keskustelualueilla. Tätä verkkokeskustelua ei kuitenkaan ole ohjattu tai arvioitu. Opettajat ovat keränneet kursseilta säännöllisesti opiskelijapalautetta harjoitustehtävien muodossa. Opiskelijoille on myös järjestetty joitakin kurssien ulkopuolisia testejä sekä kyselyjä, mutta nämä ovat liittyneet lähinnä erilaisiin ViSCoS-opiskeluun ja työvälineisiin sekä ohjelmistoihin liittyviin tutkimuksiin, eikä niitä ole käytetty varsinaisessa opiskelijoiden arvioinnissa.

6 Arvioinnin kehittäminen ViSCoS-opinnoissa

Edellisissä luvuissa käsittelin arvioinnin teoriaa, arvioinnin haasteita verkko-opinnoissa sekä ViSCoS-opintoja. Tässä luvussa vaihdan näkökulmaa ja yhdistän aikaisemmin esitellyn teorian ja tarkastelen tarkemmin kuinka arviointia voidaan kehittää ViSCoS-ympäristössä. Onnistunut arviointi määrittää pitkälti sen, mitä opiskelijat muistavat opetuksessa käsitellyistä asioista. Arvioinnilla on myös suuri vaikutus siihen, miten positiivinen mielikuva opiskelijoille jää opinnoista. Opintojen arvioinnin kehittämiseen on siis syytä panostaa.

ViSCoS-opinnot on jaoteltu kursseihin, joten on luontevinta lähteä kehittämään arviointia kurssikohtaisesti. Kullekin kurssille parhaiten soveltuva arviointi riippuu opiskelijoiden ja oppimisympäristön lisäksi kurssilla käsiteltävistä asioista. Suurin osa nykyisistä ViSCoS-kursseista arvioidaan perinteisen tentin avulla. Perinteisiin tentteihin liittyy arviointimenetelmänä kuitenkin suuri kaksoisrakentellisuuden riski, kuten luvusta 2.2 ilmenee.

Luvussa 2.6 käyn läpi sitä, millaisia ominaisuuksia hyvältä ViSCoS-kurssin arviointimenetelmältä vaaditaan. Tämän jälkeen tarkastelen nykyisten kurssien arvioinnin luvussa 6.2. Perinteisten tenttien käyttäminen ViSCoS-kursseilla velvoittaa opiskelijat hakeutumaan tenttipaikkoihin, joissa arviointi suoritetaan. Luvussa 6.3 käsittelen sitä, miksi ViSCoS-opinnoissa olisi syytä siirtyä täysin verkkopohjaiseen arviointiin.

Luvussa 6.4 esittelen suosittamani uudet arviointimenetelmät ViSCoS-kursseille ja tarkastelen, mitä etuja ja haittoja uusien menetelmien käyttöönottamisesta kursseilla on. Luvussa 6.5 käyn vielä yhteenvedon omaisesti läpi esitetyt arviointimenetelmät sekä tarkastelen palautteen kehittämistä ViSCoS-kursseilla.

6.1 Hyvän ViSCoS arviointimenetelmän ominaisuudet

Luvussa 2.6 listasin hyvän arviointimenetelmän ominaisuuksia yleisellä tasolla. Näiden yleisten periaatteiden lisäksi ViSCoS-opintojen arvioinnissa tulee ottaa huomioon sekä opetettavat asiat että opintojen verkko-opetusluonne. Kaksoisrakentellisuuden sisältyminen ei kuulu hyvään arviointimenetelmään ja sen karsimiseen ViSCoS-kurssien arvioinnista tuleekin kiinnittää erityistä huomiota. Esittelen seuraavassa joukon vaatimuksia, jotka hyvän ViSCoS-kurssin arvioinnin tulee täyttää.

1. Arviointi antaa mahdollisimman tarkan kuvan siitä, kuinka hyvin opiskelija osaa kurssin asiat.
2. Vilpin esiintyminen on mahdollisimman epätodennäköistä ja vilppi on tunnistettavissa kohtuullisella vaivalla.
3. Arviointi on mahdollisimman nopeaa ja helppoa sekä opettajalle että opiskelijalle.
4. Arviointi on niin ajasta ja paikasta riippumatonta kuin mahdollista.
5. Arviointikokemus on opiskelijalle mahdollisimman positiivinen ja arviointi on niin avointa kuin mahdollista.
6. Arviointi noudattaa luvussa 2.6 esiteltyjä hyvän arvioinnin periaatteita.

Vaatimuksista ensimmäinen liittyy siihen, että ViSCoS-opintojen arvioinnin pääasiallisena tavoitteena on saada realistinen kuva opiskelijoiden tiedoista ja taidoista kullakin kurssilta. Tätä tietoa tarvitaan sekä arvostelun tekemiseksi – eli arvosanojen antamiseksi – että opettajan seuratessa sitä, miten opetuksessa tulisi edetä. Toinen vaatimus nojaa suoraan lukuun 3.2. Valitettavasti vilppi ei ole epätavallista myöskään ViSCoS-opinnoissa, joten sen vähentäminen ja tunnistaminen on tärkeää.

Kolmas vaatimus tulee tämän tutkielman motivaatiosta. Jotta opettajien resursseja voidaan vapauttaa arvioinnista muuhun työhön, arvioinnin olisi oltava nykyistä helpommin toteutettavissa. Lisäksi arvioinnin kuormittavuutta tulee tarkastella myös opiskelijan näkökulmasta. Neljäs vaatimus on esitetty, koska ViSCoS-opintoja pyritään tarjoamaan mahdollisimman pitkälti verkko-opintoina. Olisi erittäin hyvä, että myös ViSCoS-opintojen arviointi onnistuisi mahdollisimman pitkälti verkon välityksellä.

Viides vaatimus perustuu sille, että opiskelijalla pyritään ViSCoS-opinnoissa tarjoamaan mahdollisimman positiivinen oppimiskokemus. On tärkeää, että opiskelija kokee häneen kohdistuvan arvioinnin olevan oikeudenmukaista, läpinäkyvää ja oikeutettua. Karjalainen (2001) pitää tällaista avointa arviointia eettisesti hyväksyttävän arvioinnin peruslähtökohtina.

Viimeisellä vaatimuksella lisään arviointimenetelmien tarkasteluun luvussa 2.6 esiteltyt hyvän yliopistollisen arviointimenetelmän ominaisuudet sellaisina kun Karjalainen ja Kempainen (1994) ne esittävät. Myös ViSCoS-opinnot ovat yliopisto-opintoja ja

näiden ominaisuuksien tulisi löytyä myös ViSCoS-kursseilla käytettävistä arviointimenetelmistä. Hyvä arviointimenetelmä sisältää mahdollisimman vähän kaksoisrakenteisuutta. Nämä arviointimenetelmien ominaisuudet pyrkivätkin rajaamaan kaksoisrakenteisuuden pois käytettävistä arviointimenetelmistä.

Ideaalinen ViSCoS-kurssin arviointi täyttää mahdollisimman tyydyttävästi jokaisen edellä esitetyn vaatimukseen. Todellisuudessa vaatimukset ovat ristiriitaisia jopa siinä määrin, että arviointimenetelmää, jossa kaikki vaatimukset tulevat täytetyksi, ei tällä hetkellä ole olemassa. Tässä suhteessa ideaalisen arviointimenetelmän muodostaminen ViSCoS-kursseille on mahdotonta. Tämä ongelma nousee esiin erityisesti ajasta ja paikasta riippumattoman arvioinnin ja vilpin välttämisen suhteen.

Jos halutaan olla täysin varmoja siitä, että vilppiä ei synny, luvun 3.2 perusteella voidaan vetää se johtopäätös, että verkon välityksellä tapahtuva arviointi on aina alttiina vilpille. Toisaalta tämä koskee kaikkia verkko-opintoja. Pessimistinen näkemys, jonka mukaan koskaan ei voida olla varmoja koska opiskelija tekee vilppiä verkon välityksellä (Simon, 2005), pitää paikkansa. Myös se vastaavatko opiskelijat itse omiin harjoitustehtäviinsä, voidaan helposti asettaa kyseenalaiseksi. Todellisuudessa tästä ei voida saada verkon välityksellä koskaan täyttä varmuutta. Tässä suhteessa joudutaankin tekemään kompromissi.

Myös ensimmäiseen vaatimukseen täyttäminen tyydyttävästi on hankalaa. Kuinka on mahdollista laatia yksittäiselle kurssille sellainen arviointimenetelmä, että tulos antaa mahdollisimman tarkan kuvan opiskelijan osaamisesta? Todellisuudessa ei ole välttämätöntä saada täysin kattavaa kuvaa siitä, kuinka hyvin opiskelija on oppinut kaikki kurssin asiat. Yksittäisen kurssin tapauksessa arvioinnin onnistumiseksi riittää se, että opiskelijan osaamisesta saadaan kyllin kattava kuva suhteessa kurssin sisältöön sekä kurssilla tärkeinä pidettyihin asioihin.

Yksittäisiä kursseja ja arviointimenetelmiä tarkastellessa näitä vaatimuksia ei voidaakaan pitää absoluuttisina vaan ainoastaan suuntaa antavina. Hyvä arviointimenetelmä täyttää mahdollisimman tyydyttävästi kaikki edellä listatut vaatimukset. Näitä vaatimuksia voidaan käyttää minkä tahansa arviointimenetelmän tai hajautetun arvioinnin tarkasteluun.

6.2 ViSCoS-kursseilla nykyisin käytettävät arviointimenetelmät

Tarkastelen seuraavaksi ViSCoS-kurssien nykyistä arviointia kriittisesti. Nostan esille kunkin kurssin keskeisen tietosisällön ja käyn läpi kuinka arviointi käytännössä suoritetaan. Käytän tarkastelun lähtökohtana edellisessä luvussa esille nostamiani vaatimuksia hyvälle ViSCoS-kurssien arvioinnille.

6.2.1 Tietotekniikan perusteet

Tietotekniikan perusteet -kurssilla opiskelijat tutustuvat tietotekniikan perusteisiin ja perustyövälineisiin. Kurssi on hyvin teoriapainotteinen. Viikoittaiset harjoitustehtävät pohjautuvat Moodlessa olevaan verkkomateriaaliin. Opiskelijan on suoritettava kolmasosa harjoitustehtävistä läpäistäkseen kurssin. Kurssi arvioidaan lopussa suoritettavalla perinteisellä tentillä. Varsinaisen tentin lisäksi kurssiin kuuluu HTML-osio, jossa opiskelijat harjoittelevat verkkosivun laatimista HTML-kielen avulla. Käytännössä HTML-osio on kurssin sisällä oleva pieni harjoitustyö. HTML-osio arvioidaan hyväksytty/hylätty asteikolla.

Kurssilla käsiteltävät asiat ovat sisäisesti melko epäyhtenäisiä ja tenttiin valmistautuminen muistuttaa nykyisin ulkoa opettelua. Kurssimateriaali sisältää paljon asiatietoa ja läpi käytäviä asioita on runsaasti. Kurssin oppimistavoitteiden kannalta on kuitenkin tärkeämpää pääasioiden, kuten esimerkiksi tietoverkkojen toiminnan opiskelu ja ymmärtäminen. HTML-kielen käyttöön liittyvä harjoitustehtävä harjaannuttaa opiskelijat laatimaan WWW-sivuja kielen avulla.

Nykyisellä arviointimenetelmällä opiskelijoiden oppimisesta saadaan tietoa harjoitustehtävien, tentin sekä HTML-osion perusteella. Opiskelijan suoritus arvostellaan nykyisin tentin perusteella, sillä harjoitustehtävistä saatavilla bonuspisteillä ei ole merkittävää vaikutusta opiskelijan arvosanaan. Todellisuudessa viikoittaiset harjoitustehtävät antavat huomattavasti paremman kuvan siitä, miten opiskelija on kurssilla käsitellyt asiat ymmärtänyt.

Vilppiä kurssilla voi tapahtua opiskelijan plagioidessa harjoituksia tai HTML-osiossa toisia verkkosivuja. Myös tentissä opiskelija voi toimia vilpillisesti. Vilpin tekemisen todennäköisyyttä on vaikea arvioida. Vilpin tunnistamiseen ei ole käytettävissä muita keinoja kuin opettajan tarkkaavaisuus.

Nykyisen mallin mukainen arviointi on opettajan kannalta työläs tenttien laatimisen ja korjaamisen osalta. Opiskelijan kannalta perinteiseen tenttiin valmistautuminen on myös raskasta. HTML-osio on opettajan kannalta nopea arvioitava. Suurin osa opettajan työajasta menee viikoittaisten harjoitusten arviointiin ja palautteen antamiseen. Tämä palaute on kuitenkin opiskelijan kannalta erittäin tärkeää, kuten Pölönen (2005) toteaa.

Arvioinnin ajoitus on sidottu viikoittaisiin harjoitustehtäviin, sekä tenttiaikatauluihin. Opettajan kannalta tämä tarkoittaa sitä, että tentit pitää valmistella ja tarkastaa kiireisellä aikataululla postitusaikojen vuoksi. Opiskelija taas joutuu matkustamaan tenttitilaisuuteen, eikä hän voi suorittaa tenttiä kuin ennalta määrättyssä paikassa.

Opiskelijan kokemus arvioinnista painottuu varmasti pitkälti perinteiseen tenttiin, joka päättää kurssin. Suurelle osalle opiskelijoista tentti on stressaava kokemus. Opiskelijat saavat tietää tenttituloksensa sekä tehtäväkohtaisen pisteytyksen. Opiskelijoilla ei kuitenkaan ole mahdollisuutta saada palautetta suorituksestaan tai tietää, mitä asioita he tentissä hallitsivat ja mitä eivät. HTML-osion palaute on avointa.

Perinteiseen tenttiin liittyy kaksoisrakenteen ongelma, joka on esitelty tarkemmin luvussa 2.2. Kurssilla käytettävä arviointi vastaa huonosti hyvän yliopistollisen arviointimenetelmän periaatteita. Kurssin aikana tehtävät harjoitukset kyllä valmentavat tiedon etsimiseen ja löytämiseen, mutta näitä taitoja ei voi hyödyntää loppuarviointitilaisuutena toimivassa perinteisessä tentissä. Arviointimenetelmä ei valmenna yhteistyöhön, eikä oman työn arviointiin.

6.2.2 Ohjelmointi 1

Ohjelmointi 1 -kurssi on ensimmäinen ViSCoS-opintojen ohjelmointikurssi. Kursin tavoitteena on perehdyttää opiskelijat Java-ohjelmointiin ja ohjelmoinnin kontrollirakenteisiin sekä perustietotyyppeihin. Kurssin aikana opiskelijat harjoittelevat Java-ohjelmointia verkkomateriaalin, oppikirjan ja viikoittaisten harjoitustehtävien avulla. Viikoittaisten harjoitustehtävien yhteydessä opiskelijoilta kerätään itsearviointi, jota ei kuitenkaan arvostella. Harjoitustehtävistä opiskelijat saavat bonuspisteitä tenttiin. Kurssi arvioidaan lopuksi suoritettavalla perinteisellä tentillä.

Kurssin arviointi vastaa Tietotekniikan perusteet kurssin arviointia sillä erotuksella, että kurssilla opiskellaan teorian sijasta ohjelmointia ja kurssilla ei ole HTML-osuutta vastaavaa harjoitustyötä. Edellä esitetyt arviot Tietotekniikan perusteet -kurssin arviointikäytännön ongelmista pätevät myös Ohjelmointi 1 -kurssille.

Kurssin arviointia leimaa kuitenkin vielä edellisessä luvussa käsiteltyä kurssia voimakkaampi kaksoisrakenne. Perinteinen tentti vastaa hyvin vähän todellista ohjelmointitilannetta. Tätä ongelmaa tarkastelen tarkemmin luvussa 3.3. Ohjelmoinnin oppimisen kannalta perinteisen tentin ongelmat korostuvat. Opiskelijan kannalta erittäin valitettavaa nykyisessä käytännössä on se, että opiskelija ei saa palautetta menestymisestään tentissä, joten hänen on vaikea tietää, millä perusteella arviointi on tehty ja miten hänen ohjelmointitaitonsa on kehittynyt.

6.2.3 Tietojenkäsittelytieteen tutkimusaloja

Tietojenkäsittelytieteen tutkimusaloja -kurssilla opiskelijat tutustuvat johonkin tietojenkäsittelytieteen tutkimusalaan ja laativat siitä yleensä ryhmätyönä lyhyehkön essee-tyyppisen kirjoitelman. Kurssilla tutustutaan myös tieteelliseen kirjoittamiseen ja viittaustekniikkaan. Opiskelijat laativat kurssin päätteeksi myös oppimispäiväkirjan, johon liittyy muiden ryhmän jäsenten arviointi sekä kolmen muun ryhmän laatiman kirjoitelman kirjallinen arviointi. Tietojenkäsittelytieteen tutkimusaloja -kurssi arvosellaan tällä hetkellä kirjoitelman perusteella sekä oppimispäiväkirjan avulla.

Käytettävät arviointimenetelmä antaa melko hyvän kuvan siitä, miten ryhmä hallitsee tieteellisen kirjoittamisen. Sen sijaan yksilösuorituksia kirjoitelmasta on vaikea arvioida. Oppimispäiväkirja ja etenkin sen vertaisarviointiosuus paljastaa kuitenkin opettajalle miten ryhmätyön tekeminen on sujunut ja onko työ jakaantunut tasaisesti.

Vilpin riski kurssilla liittyy lähinnä tutkielmatekstin plagiointiin Internetistä. Tämä riski on olemassa, sillä valmis Internetissä julkaistu teksti voi helposti houkuttaa opiskelijat käyttämään sitä osana kirjoitelmaa ilman lähdeviitteitä. Vilpin tunnistaminen on opettajan valppauden varassa. Arviointi on opettajan kannalta haastavaa, sillä yksilöiden suorituksia on vaikea arvioida. Oppimispäiväkirjaan liittyvät vertaisarvioinnit helpottavat kuitenkin opettajan arviointityötä.

Kurssin arviointimenetelmät mahdollistavat sen, että opiskelija voi työstää työtä missä tahansa, eikä arviointia ole sidottu yksittäiseen paikkaan. Opettajalla taas on mahdollisuus käyttää runsaasti aikaa tutkielmien ja oppimispäiväkirjojen arvosteluun. Käytännössä arviointi niin ajasta ja paikasta riippumaton kun se on mahdollista. Kurssin arviointimenetelmät ovat opiskelijan kannalta varsin avoimia. Opettaja arvostelee suoritukset sekä sanallisesti että antamalla arvosanan. Opiskelija saa perustellun palautteen työstä, sekä omasta suoriutumisestaan.

Tietojenkäsittelytieteen tutkimusaloja -kurssilla täytössä oleva arviointimenetelmä vastaa nyt varsin hyvin hyvän yliopistollisen arvioinnin periaatteita. Ryhmätyöskentely ja tiedonhaku ovat osa tutkielman työstämistä. Opiskelijalla on mahdollista harjoitella kurssin aikana myös muiden ryhmän jäsenten arviointia.

6.2.4 Ohjelmointi 2

Ohjelmointi 2 -kurssi on ViSCoS-opintojen toinen varsinainen ohjelmointikurssi. Kurssilla opiskelijat tutustuvat laajemmin Java-kieleen. Kurssin pääaiheita ovat olio-ohjelmointi, Java-kielen luokkakirjastot sekä graafisen käyttöliittymän ohjelmointi Javalla. Laajuudeltaan kurssi on hieman suurempi, kuin edellinen ohjelmointikurssi. Kurssi vastaa suoritus- ja arviointimenetelmiltään Ohjelmointi 1 -kurssia. Myös arviointiin liittyvät ongelmat vastaavat täysin Ohjelmointi 1 -kurssia.

6.2.5 Perusopintojen harjoitustyö

Perusopintojen harjoitustyö -kurssilla opiskelijat laativat itsenäisesti opettajan ohjauksella laajemman Java-ohjelman ja testaavat sekä dokumentoivat työnsä. Yleensä kurssilla tuotettu ohjelma on opiskelijoiden ensimmäinen itse ohjelmoitu laajempi tietokoneohjelma. Kurssin opettaja arvostelee valmiin ohjelman sekä siihen liittyvän dokumentoinnin ennalta määriteltyjen arviointikriteerien perusteella. Sekä ohjelmaa että dokumentointia painotetaan arvioinnissa yhtä paljon.

Koska opettaja arvioi harjoitustyönä syntynyttä ohjelmaa ja dokumenttia, hän saa hyvin tietoa opiskelijan taitotasosta. Harjoitustyötä arvioitaessa opettaja pystyy työn perusteella päättelemään opiskelijan ohjelmointitaidon ja kyvyn laatia teknistä dokumentaatiota. Vilpin riski kurssilla liittyy plagiointiin. Valmista lähdekoodia on helppo lada-

ta Internetistä ja käyttää osana työtä. Vilpin tunnistaminen on tällä hetkellä opettajan valppauden varassa.

Kurssin suurin ongelma liittyy arvioinnin työtaakkaan opettajan kannalta. Sama harjoitustyö saattaa käydä opettajan tarkastettavana jopa viisi kertaa ennen lopullista hyväksymistä. Jokaisella arviointikerralla opettajalla on luettavanaan useita satoja rivejä ohjelmakoodia ja kymmeniä sivuja dokumenttia. Luvussa 3.3 mainitut ohjelmakoodin arviointiin liittyvät ongelmat nousevat esille myös harjoitustöitä arvioitaessa. Käytännössä opiskelijoille on annettu mahdollisuus palauttaa työ uudelleen niin monta kertaa, että opettaja voi sen hyväksyä.

Arviointi harjoitustyön laadusta sekä ohjelman että dokumentaation osalta annetaan kirjallisesti ja koko työ arvostellaan numeerisesti. Opiskelijan kannalta arviointi avointa. Varsinaisen harjoitustyön tekemisen kuormittavuus riippuu pitkälti siitä, millaiset ohjelmointitaidot opiskelijalla on. Hyvälle ohjelmoijalle työ on helppo, mutta heikolla ohjelmointitaidolla työn tekeminen on hyvin hidasta, sillä tarvittavien taitojen pitää karttua työtä tehtäessä.

Arviointi vastaa nyt hyvän yliopistollisen arviointimenetelmän ominaisuuksia. Harjoitustyön aikana opiskelijat harjoittelevat ohjelmointitaitoja ja varsinainen ohjelmointi- ja dokumentointityö vastaa hyvin pitkälti opiskelijoiden harjoitustyön aikaista työskentelyä. Tällä hetkellä harjoitustyössä ei kuitenkaan ole ryhmätyöelementtejä, mutta niiden mukaan ottaminen ei ole tämän kurssin osalta tarpeellista, sillä ryhmätyö lisäisi harjoitustyön kuormittavuutta, mutta toisi sen tekemiseen hyvin vähän etuja.

6.2.6 Diskreetit rakenteet

Diskreetit rakenteet -kurssi on varsin uusi. Se otettiin mukaan ViSCoS-opintoihin vasta syksyllä 2007. Kurssi poikkeaa muista ViSCoS-kursseista siten, että opetettava asia on pääasiassa matematiikkaa. Kurssilla käsitellään tietojenkäsittelytieteen ymmärtämisen kannalta tärkeitä diskreetin matematiikan osa-alueita. Kurssi arvostellaan loppuentillä. Kuten muillakin kursseilla, jotka arvostellaan perinteisen tentin avulla, kurssilla on harjoitustehtäviä, joista opiskelijat saavat tenttiin bonuspisteitä.

Kurssilla käsiteltävät asiat poikkeavat melko paljon muilla ViSCoS-kursseilla käsitellyistä asioista. Matematiikan soveltamisen arviointiin perinteinen tentti soveltuu paljon

paremmin kuin esimerkiksi ohjelmointikurssille. Jonkin verran kaksoisrakenteisuutta tenttiin kuitenkin tälläkin kurssilla liittyy. Perinteisen tentin käyttäminen kurssilla on kuitenkin ongelmallista tenttien tarkastukseen liittyvien aikarajoitusten ja tenttipapereiden postitusaikojen suhteen.

6.2.7 Tietokoneen rakenne ja käyttöjärjestelmä

Tietokoneen rakenne ja käyttöjärjestelmä -kurssi vastaa muita loppuentillä suoritettavia kursseja. Kurssin päätarkoituksena on tutustuttaa opiskelijat tietokoneen toimintaan ja tietokoneen rakenteeseen liittyviin fysikaalisiin lainalaisuuksiin. Loppuenttiin liittyy kurssilla kaksoisrakenteisuutta, sillä harjoitustehtävät vaikuttavat vain vähän lopulliseen arviointiin ja kurssin loppuentti muistuttaa ulkomuistikoetta.

6.2.8 Johdatus tietojenkäsittelyn etiikkaan

Johdatus tietojenkäsittelyn etiikkaan -kurssia ei arvostella, vaan kurssin ajatuksena on tutustuttaa opiskelija tietotekniikkaan liittyviin eettisiin ongelmiin. Opiskelijat laativat kurssilla essee-tyyppisen kirjoitelman jostain tietotekniikan hyödyntämiseen liittyvästä eettisestä ongelmasta. Kirjoitelma katsotaan hyväksytysti suoritetuksi, kun se on laadittu kurssin kirjoittamisohjeita noudattaen ja kirjoitelma sisältää sopivassa suhteessa sekä kirjoittajan omia että lähdekirjallisuudesta ammennettuja mielipiteitä. Kurssin lopuksi opiskelijat kirjoittavat oppimispäiväkirjan, jossa he analysoivat omaa oppimistaan ja suhtautumistaan kurssilla käsiteltyihin asioihin. Kurssi poikkeaa muista ViSCoS-kursseista siinä, että kurssista ei anneta numeroarviointia, vaan se arvostellaan skaalalla hyväksytty/hylätty.

Kurssin arviointi perustuu opiskelijan laatimaan esseeseen ja oppimispäiväkirjaan. Kirjoitelma ja oppimispäiväkirja antavat hyvän kuvan opiskelijan kyvystä laatia kirjoitelma sekä eritellä tietoa. Oppimispäiväkirja avaa lisäksi opiskelijan kirjoitusprosessia. Suurin vilpin tekemisen riski liittyy plagiointiin. Opiskelija voi kopioida valmista Internetissä julkaistua tekstiä osaksi työtään. Vilpin tunnistaminen riippuu opettajan valpaudesta.

Kurssin arviointi on opettajan kannalta nopeaa, sillä työstä ei tarvitse antaa numeerista arvioita vaan pelkästään kirjallinen palaute. Arviointi on opiskelijan kannalta riippu-

matonta suorituspaikasta. Opettajalla on nykyisen opiskelijan oikeusturvaa koskevan johtosäännön (Joensuun yliopisto, 2006) mukaan kuukausi aikaa tarkastaa opiskelijan työ, joka riittää mainiosti, sillä essee palautetaan Moodleen, eikä sitä tarvitse liikutella postitse.

Arviointi noudattaa hyvän yliopistollisen arvioinnin periaatteita lukuun ottamatta ryhmätyöosuutta. Kurssin laajuuden huomioon ottaen ryhmätyön mukaan ottaminen kursseille ei ole kuitenkaan järkevää.

6.3 Perinteisistä tenteistä verkkoarviointiin

Luvussa 2.6 luetellut hyvien arviointimenetelmien ominaisuudet eivät kuvasta perinteistä tenttiä tai muutakaan ulkomuistiin nojaavaa arviointimenetelmää. Opiskelijan eristäminen opintosuorituksen aikana normaalista toimintaympäristöstä ei usein ole arvioinnin kannalta järkevää, koska tällöin ajaudutaan väistämättä kaksoisrakenteelliseen arviointiin, jollei tarkoituksena ole nimenomaan harjoitella tällaisessa tilanteessa toimimista. Ainoa peruste opiskelijan eristämiseksi on se, että voidaan kontrolloida arvioinnin kohdistuvan juuri tiettyyn opiskelijaan ja siihen, toimiiko hän itsenäisesti.

Verkko-opinnoissa opiskelijan todellista työskentelyä ei voida seurata ja valvoa samoin kuin kontaktiopinnoissa. Myös vilpin tekemisen riski kasvaa verkko-opinnoissa suhteessa kontaktiopintoihin. Tästä huolimatta verkko-opinnoissa on pystyttävä luottamaan opiskelijaan siinä määrin, että opiskelijan palauttamat harjoitukset ja harjoitustyöt arvioidaan opiskelijan omina suorituksina, jollei ole syytä epäillä vilppiä. ViSCoS-opinnot on pyritty tarjoamaan mahdollisuuksien mukaan kokonaan verkko-opintoina. Tällä hetkellä ainoa kontaktiopetusmuoto ovat perinteiset tentit.

Koska opinnot suoritetaan joka tapauksena verkko-opintoina ei vilpin tekemisen riskiä voida eliminoida nykyisessä käytännössä kuin kurssien loppuarviointitilanteissa. Jos nykyinen malli, jossa voidaan todeta opiskelijoiden osallistuminen henkilökohtaisesti arviointitilaisuuteen halutaan säilyttää, arviointia tulisi lähteä kehittämään tältä pohjalta. Vaihtoehtoisia arviointimuotoja hyödynnettäessä tämä vaatisi kuitenkin, että opiskelijoilla on arviointitilaisuudessaan käytössään tietokone ja verkkoyhteys. Tämä voi olla hankalaa järjestää varsinkin kehittyvissä maissa. Tällaista arviointia hyödynnettäessä opiskelijat joudutaan velvoittamaan myös matkustamaan arviointitilaisuuteen.

Valvottuja loppuarviointitilaisuuksia hyödyntävien arviointimenetelmien kehittäminen on mahdollista, mutta se ei nähdäkseni ole saavutettuun hyötyyn suhteutettuna mielekäs tapa arvioinnin kehittämiseen ViSCoS-opintojen näkökulmasta. Verkko-opetuksessa joudutaan loppuarviointimenetelmästä huolimatta luottamaan siihen, että opiskelija toimii kurssilla itse. Vaikka Simon (2005) pitääkin tällaista luottamusta opiskelijoihin naiivina, se on ainoa tapa jolla verkko-opintoja voidaan ylipäätään perustella.

Arviointimenetelmien valinnassa painotankin opintojen ajasta ja paikasta riippumattomuutta enemmän, kuin vilpillisen toiminnan riskiä. Vilpin esiintyminen on tosiasia, eikä vilppiä pidä hyväksyä, mutta myöskään näkökulma, jossa pelätään jokaisen opiskelijan työn olevan vilpillisesti toteutettu, ole mielekäs opiskelijan, opettajan eikä arvioinnin kannalta. Tarvittaessa plagioinnin tunnistamiseen käytettävät ohjelmistot voidaan integroida osaksi ViSCoS:ssa käytössä olevaa Moodlea. Tällöin mahdollisten plagiointitapausten ilmenemisen todennäköisyys kasvaa.

Eräs mahdollisuus varmistaa se, että opiskelija omaksunut ainakin jotakin tietoja opintojen aikana, on haastatella jokainen opiskelija puhelimitse ennen kokonaissuoritusmerkinnän antamista ViSCoS-opinnoista. Haastattelun voi hoitaa siten, että opiskelija velvoitetaan osallistumaan haastatteluun nykyistä tenttipaikkaa vastaavassa tilassa, jossa hänen henkilöllisyytensä voi varmistaa joku luotettava henkilö. Varsinaisessa haastattelussa voidaan kysyä tarkentavia satunnaisia kysymyksiä opiskelijan suorituksista opintojen aikana, opiskelijan mielipiteitä kursseista sekä oppimateriaalista ja kerätä kurssipalautetta. Tämä malli vastaa Thomas *et al.* (2001) esittämää menetelmää, mutta yksittäisen kurssin sijaan haastattelu tehtäisiin koko opinnoista ja opiskelijat tunnistettaisiin luotettavasti.

Kuten luvussa 3.2.1 on todettu, vilpin tekemisen todennäköisyyttä saadaan vähennettyä, mikäli opiskelijat kokevat opiskelun mielekkääksi ja ottavat vastuuta omista opinnoistaan. Vastuullisuutta opiskeluun saadaan lisäämällä esimerkiksi harjoitustehtävien painotusta kurssien arvioinnissa. Opiskelijoiden vastuullisuutta voidaan myös lisätä ottamalla heidät mukaan arviointiprosessiin kuten Karjalainen ja Kemppainen (1994) ehdottavat.

6.4 Suositellut arviointimenetelmät ViSCoS-kursseille

Lähes kaikkiin ViSCoS-kursseihin liittyy luvun 6.2 olevan tarkastelun valossa ongelmia. Erityisen ongelmallisia ovat Ohjelmointi 1 ja Ohjelmointi 2 -kurssit niiden voimakkaan kaksoisrakenteen takia. Ongelmat eivät kuitenkaan rajoitu pelkästään näihin kursseihin. Kaksoisrakenteen havaittavissa myös muilla perinteisellä tentillä arvioitavissa kursseilla. Arviointimenetelmiä kehitettäessä tulee etsiä keino kaksoisrakenteen purkamiseen ja löytää mielekkäät kullekin kurssille soveltuvat arviointitavat.

Arviointimenetelmien valinta ViSCoS-kursseille ei ole täysin yksiselitteistä. Luvussa 6.1 esitettyjen vaatimusten pohjalta voidaan tarkastella sitä, millainen on hyvä ViSCoS-kurssille soveltuva arviointimenetelmä. Koska ViSCoS-kurssit ovat kohtuullisen laajoja kokonaisuuksia, luvussa 4 esitellyt yksittäiset arviointimenetelmät soveltuvat huonosti ainoana arviointivälineenä kursseille. Kurssien arviointi tuleekin hajauttaa useampaan osakokonaisuuteen, joista muodostuu kurssin kokonaisarviointi.

ViSCoS-kursseilla opiskelijoilla teetetään paljon harjoituksia, jotka arvostellaan sekä sanallisesti että numeerisesti. Karjalainen ja Kemppainen (1994) suosittelevat, että mallintavat arviointimenetelmät, joihin harjoitustehtävät kuuluvat, on syytä ottaa käyttöön kokonaisarvioinnissa. Kirjoittajien mukaan tällaisten mallintavien arviointimenetelmien hyöty valuu hukkaan, mikäli kurssilla järjestetään harjoitusten lisäksi perinteinen tentti. Harjoitustehtävät voidaankin luontevasti ottaa osaksi kurssien arviointia ja siirtää arvioinnin painoa — sekä opiskelijoiden keskittymistä — kaksoisrakenteelliselta tentiltä kurssin harjoitustehtäviin. Harjoitustehtävät ja erityisesti niistä saatava palaute ovat myös tärkeitä opiskelijoille (Pölönen, 2005).

Esitän seuraavissa luvuissa erään mahdollisuuden kurssien arvioinnin kehittämiseen. Arviointimenetelmien valinta perustuu omaan kokemukseeni ViSCoS-kurssien opetuksesta sekä kirjallisuudesta löytyvään tietoon tietojenkäsittelytieteen verkkokurssien arvioinnista. Kurssien eri arviointimenetelmien väliset painotukset ovat esimerkiksi omaisia. Lopullisten painotusten valinta riippuu siitä, minkälaisia asioita kurssien kokonaisarvioinnissa halutaan painottaa.

6.4.1 Tietotekniikan perusteet

Tietotekniikan perusteet -kurssilla käsiteltävät asiat ovat sirpalemaisista ja nykyiseen perinteiseen tenttiin osallistuvalla oletetaan suuren tietomäärän ulkoa muistamista. Kurssin oppimistavoitteiden kannalta ei kuitenkaan ole tarkoituksenmukaista olettaa opiskelijoiden opettelevan kurssilla käsiteltäviä asioita ulkoa, vaan tutustuttaa tietojenkäsittelytieteen alaan ja tietotekniikkaan sekä tietoverkkoihin. Opiskelijat palauttavat kurssilla viikoittaisia harjoitustehtäviä. Koska harjoitustehtävät arvioidaan tälläkin hetkellä, olisi luontevaa ottaa nämä harjoitustehtävät suuremmaksi osaksi kurssin arviointia.

Nykyisen arviointitavan mukaan opiskelijan on tehtävä kolmasosa kurssin harjoitustehtävistä hyväksytysti voidakseen osallistua tenttiin. Pakollisten harjoitustehtävien määrää voisi nostaa esimerkiksi puoleen kurssin kaikista tehtävistä. Kaikkien opiskelijan hyväksytysti suorittamien harjoitustehtävien arvostelu voisi muodostaa esimerkiksi puolet kurssin kokonaisarvostelusta. Kurssiin sisältyvä pakollinen HTML-osuus, joka arvostellaan arvosanalla hyväksytty/hylätty on syytä säilyttää kurssilla, sillä siinä harjoiteltavia taitoja tarvitaan myöhemmissä opinnoissa.

Anewalt (2002) esittää, että kirjoitelmien käyttäminen tietojenkäsittelytieteen kursseilla aktivoi opiskelijoiden kriittistä ajattelua ja kirjoitelmien hyödyntämistä tietojenkäsittelytieteen opinnoissa tulisi lisätä. Myös The Joint Task Force on Computing Curricula (2001) suosittelee viestintätaitojen kehittävien opintojen lisäämistä opetuksessa. Loppuosan arvostelusta voisi kurssin sisällön huomioon ottaen kattaa jotakin kurssilla käsiteltäviä asioita syventävällä esseellä. Mahdollisia esseen aiheita voisivat olla esimerkiksi TCP/IP-protokolla, Internetin rakenne, hyvä työergonomia, tietoturva, käyttöjärjestelmät jne. Hyvin kirjoitettu esse perehdyttää opiskelijan paremmin yhteen kurssilla käsiteltyyn osa-alueeseen, kun taas harjoitustehtävillä varmistutaan siitä, että opiskelijat tutustuvat koko kurssimateriaaliin.

Ehdotettu arviointimalli poistaa kokonaan tentit kurssin arvostelusta, joten tenttiin liittyvä kaksoisrakenne saadaan vältettyä. Tällaisen arvioinnin kuormittavuutta opettajan kannalta on hankala arvioida luotettavasti. Työmäärä riippuu pitkälti siitä, miten laajoja esseitä opiskelijoilla teetetään. Käytännössä esseen arviointiin käytettävän ajan voidaan kuitenkin olettaa vastaavaan tentin tarkastukseen käytettävää aikaa. Suurin vilpin riski liittyy harjoitustehtävien ja esseen plagiointiin. Plagioinnin mahdollisuus tulee ottaa huomioon opiskelijan töitä arvosteltaessa.

Ensimmäistä kertaa uutta arviointimallia käytettäessä aikaa kuluu myös esseen aiheiden laadintaan sekä esseenkirjoitusohjeiden tekemiseen. Koska kurssilla käsiteltävät asiat ovat melko sirpaleisia, on myös syytä miettiä, onko järkevää arvostella kurssi numeroarvosanalla vai riittäisikö arvioinniksi pelkästään sanallinen arviointi ja hyväksytty/hylätty arvosana.

Esitetyn arviointimallin etuna on, että kurssin arviointi ja suoritus muuttuu ajasta ja paikasta riippumattomaksi. Uudistettuja arviointimenetelmiä käyttämällä arviointi on opiskelijan kannalta läpinäkyvämpää. Arvioinnin muuttaminen yllä esitetyn kaltaiseksi saattaa kurssin arvioinnin myös vastaamaan paremmin luvussa 2.6 esitettyjä hyviä yliopistollisen arviointimenetelmien ominaisuuksia.

6.4.2 Ohjelmointi 1

Nykyistä Ohjelmointi 1 -kurssin arviointia perinteisellä tentillä leimaa voimakas kaksoisrakenteisuus. Ohjelmointitaitoa ei voida luontevasti mitata, kun opiskelija vastaa ohjelmointikysymyksiin kynällä. Ohjelmoinnin arvioinnin problematiikkaa on tarkasteltu lähemmin luvussa 3.3.

Opiskelijan palauttamien ohjelmointivastausten automaattiseen arviointiin on kehitetty erilaisia menetelmiä, joita esittelen luvussa 3.4.3. Vaikka ohjelmoinnin automaattiseen arviointiin on kehitetty useita ohjelmistoja, ne sopivat nykyisessä muodossaan melko huonosti ViSCoS-kursseille. Käytettävissä olevat työkalut eivät sovellu graafisten ohjelmien arviointiin. Suuri osa kurssin harjoitustehtävistä on kuitenkin suunniteltu visuals first -periaatteen mukaan, ja opiskelijat laativat pääosin graafisia ohjelmia.

Automaattiset ohjelmointitehtävien tarkastukseen laaditut ohjelmat rajoittavat harjoitustehtävien rakennetta kuten Lehtonen (2002) toteaa. Myös ohjelmistojen käyttöönotto on hankalaa. Tästä huolimatta ohjelmointitehtävien arviointia helpottavia välineitä tulisi testata käytännössä ViSCoS-kursseilla, sillä niistä olisi erityisen suuri apu opettajan tarkastustyön tukena. Opiskelijat tarvitsevat ohjelmointikurssilla paljon palautetta, kuten Torvinen (2004) ja Pölönen (2005) ovat osoittaneet. Jos ohjelmointitehtävät arvioitaisiin automaattisesti, tämä mahdollistaisi välittömän palautteen antamisen. En voi kuitenkaan suositella automaattisten ohjelmakoodin arviointivälineiden käyttöä ohjelmointikurssien arviointiin ennen kun niiden toiminnasta varmistutaan ja välineet saadaan testattua kattavasti.

Koska ohjelmointi on ennen kaikkea opeteltu taito, harjoitustehtävien tekeminen kartuttaa samalla ohjelmointitaitoa. Tästä johtuen onkin luontevaa siirtää arvioinnin painotusta loppuentistä kurssin aikana suoritettaviin harjoitustehtäviin. Harjoitustehtävien tekeminen, eli ohjelmoinnin harjoittelu on ainoa tapa opetella ohjelmointia, sillä ohjelmoimaan opitaan vain ohjelmoimalla, kuten Sheil (1981) on osoittanut. Ohjelmointi 1 -kurssi on useille opiskelijoille aivan ensimmäinen kosketus ohjelmointiin. Harjoitustehtävien arvioinnin ohessa on erityisen tärkeää keskittyä samalla myös opiskelijoiden ohjaukseen, kuten Pölönen (2005) ehdottaa.

Ohjelmointi 1 -kurssin arvioinnissa olisi syytä siirtyä loppukokeen sijasta keskittyä arvioimaan opiskelijoiden ohjelmointitaidon kehitystä. Tämä onnistuu parhaiten ottamalla harjoitustehtävät osaksi kurssin lopullista arviointia. Pakollisten harjoitustehtävien vähimmäismäärää voisi nostaa nykyistä kolmasosaa suuremmaksi.

Kurssin loppukokeen sijasta voisi opiskelijoilla teettää kokoavan harjoitustehtävän, jossa kootaan yhteen kurssilla käsitellyt asiat. Tehtävän tulisi sisältää kaikki kurssin sisällön kannalta oleelliset Java-kielen kontrollirakenteet, tietotyypit jne. Opettajan arviointityön helpottamiseksi ja opiskelijan oman oppimisen kannalta olisi hyvä, jos opiskelija voisi arvioida omaa osaamistaan loppuharjoituksessa ja ehdottaa itselleen työstä oikeaksi kokemaansa arvosanaa. Viikoittaisen itsearvioinnin sijasta osana arviointia voisi toimia viikoittain täydennettävä oppimispäiväkirja.

Eräs mahdollisuus painottaa uusi arviointi on muodostaa 60 % arvosanasta viikoittaisista harjoitustehtävistä saatujen arvosanojen perusteella. Vastaavasti kokeen korvaava loppuharjoitustehtävä voisi olla painoltaan 40 % kurssin kokonaisarvosanasta. Viikoittain täydennettävä ja kurssin lopuksi yhteen koostetun oppimispäiväkirjan on helppoa arvostella hyväksyty/hylätty asteikolla. Haluttaessa myös oppimispäiväkirja voidaan kuitenkin arvostella myös numeroarvosanalla.

George (2002) on saanut hyviä kokemuksia oppimispäiväkirjan käytöstä ohjelmointikurssilla. Vastaavasti ViSCoS-opettajat ovat kokeneet Ohjelmointi 1 -kurssin itsearvioinnin hyödylliseksi sekä opiskelijoiden että itsensä kannalta. Kurssien normaalin kehittämisen lisäksi ohjelmointikurssilla täydennettävät oppimispäiväkirjat tuovat mielenkiintoista lisätietoa ohjelmoinnin oppimisesta sekä oppimateriaalin laadusta. Tätä oppimispäiväkirjatietoa voidaan käyttää hyödyksi myös tutkimustyössä.

Esitetyn arviointimenetelmän etuna on, että perinteisen paperille vastattavan ohjelmointitenttin voimakas kaksoisrakenteisuus saadaan purettua. Kun arvioinnissa siirrytään painottamaan harjoitustehtäviä, opiskelijat saadaan paremmin motivoitua harjoitustehtävien tekemiseen. Tällä tavoin kurssin arvioinnista tulee jatkuvaa arviointia. Kokoavassa harjoitustehtävässä opiskelijoilla on mahdollisuus osoittaa ohjelmointitaitonsa kehittyminen.

Työmäärällisesti ehdotettu malli vastanee opettajan näkökulmasta nykyistä tenttipohjaista arviointia. Kun uutta arviointimenetelmää otetaan käyttöön, jonkin verran lisää aikaa kuluu oppimispäiväkirjan ohjeistukseen ja kokoavan harjoitustehtävän aiheiden laatimiseen ja ohjeistamiseen. Arvioinnin osien painotukset riippuvat paljon siitä, miten laaja loppuharjoitustehtävä opiskelijoilla teetetään.

Tämän kaltaisella arviointimenetelmällä ohjelmointikurssin arviointi saadaan paljon lähemmäksi luvussa 2.6 esitettyä hyvää arviointia. Ohjelmointi on kokonaisvaltainen prosessi ja ohjelmointitaito karttuu ohjelmoimalla. Tässä suhteessa arvioinnin tuleekin keskittyä erityisesti ohjelmoititaidon kehittymisen arviointiin.

6.4.3 Tietojenkäsittelytieteen tutkimusaloja

Tietojenkäsittelytieteen tutkimusaloja -kurssi on nyt opiskelijan näkökulmasta ajasta ja paikasta riippumaton lukuun ottamatta ryhmätyöhön käytettävää ryhmän yhteistä aikaa. Kurssin keskeisenä ajatuksena on tutustuttaa opiskelijat tietojenkäsittelytieteen tutkimusalana, toimia kevyenä johdantona tieteelliseen kirjoittamiseen ja harjoittaa opiskelijoiden ryhmätyötaitoja. Nämä tavoitteet toteutuvat kohtuullisen hyvin nykyistä arviointimenetelmää käyttämällä. Kurssin arvostelu perustuu ryhmän laatimalle esseelle ja opiskelijoiden oppimispäiväkirjoille. Kuten luvussa 6.2.3 on todettu, nykyinen arviointi vastaa hyvin hyvän yliopistollisen arvioinnin periaatteita.

Kurssin arvioinnin suurin ongelma on opettajan työtaakka kurssilla. Tietojenkäsittelytieteen tutkimusaloja -kurssin arviointiprosessia voidaan jonkin verran keventää siirtymällä arvostelussa hyväksyty/hylätty arvosteluun. Kurssilla opettettavien asioiden kannalta ei ole tarkoituksenmukaista antaa opiskelijoille yksilöllisiä arvosanoja, sillä yksittäisten opiskelijoiden panostusta kurssin ryhmätyöosuuteen on opettajan näkökulmasta hyvin vaikea määritellä.

Numeroarvosana ei nähdäkseni myöskään ole välttämätön kurssin kannalta. Sanders (1988) pitääkin väkinäistä arvosanan antamista kaikista tehtävistä ainoastaan opiskelijoiden motivointikeinona, joka ei kuulu hyvään akateemiseen arviointiin. Tutkielmien ja oppimispäiväkirjojen arviointia voidaan yhdenmukaistaa käyttämällä arvioinnissa apuna luvussa 4.3 esiteltyä matriisiarviointia.

6.4.4 Ohjelmointi 2

Ohjelmointi 2 -kurssi vastaa suoritustavoiltaan ja ongelmiltaan pitkälti Ohjelmointi 1 -kurssia. Erona on, että kurssi on jonkin verran laajempi ja opiskeltavaa asiaa on kursilla enemmän. Tämänkin kurssin osalta suurin ongelma on perinteinen tentti, joka soveltuu hyvin kankeasti ohjelmoinnin osaamisen arviointiin. Arvionnin kehittämiseksi kurssilla suosittelen samankaltaista arviointimallia kuin Ohjelmointi 1 -kurssilla. Ainoana poikkeuksena on, että koska kurssi on laajuudeltaan melko paljon suurempi, yhden kokoavan laajan loppuharjoituksen sijasta on opiskelijoiden kannalta mielekkäämpää teettää kurssilla kaksi laajempaa kokoavaa harjoitustehtävää.

Ehdotettu uudistettu arviointi kurssille on siis, että kurssin harjoitustehtävien arvostelusta määrittää 60 % kurssin kokonaisarvosanasta. Kurssin puolivälissä ja lopussa palautettavat laajemmat harjoitustehtävät joiden arvostelulla olisi kummallakin 20 % paino kurssin loppuarvosanaa laskettaessa. Myös Ohjelmointi 2 -kurssin itsearviointi olisi mielekästä muuttaa oppimispäiväkirjaksi.

Koska esitetyssä mallissa on kaksi laajempaa harjoitustehtävää, opettajan kannalta kurssiin liittyvä arviointityö lisääntyy nykyiseen arviointimalliin verrattuna. Toisaalta hajauttamalla kokoava arviointi kahteen laajempaan harjoitustehtävään saadaan arviointiin käytettyä aikaa jaettua tasaisemmin. Jos kurssin asiat yhdistettäisiin yhdeksi harjoitustehtäväksi, se olisi hyvin laaja, opiskelijoille työläs tehdä ja hankala arvostella. Kurssin kannalta pääasia on kuitenkin saada muutettua arviointi perinteisestä tentistä paremmin oppimista tukevaan arviointimuotoon.

6.4.5 Perusopintojen harjoitustyö

Perusopintojen harjoitustyö -kurssin suoritusmuoto on pitkälti rajattu jo kurssin nimesä. Kurssin aikana opiskelija laatii harjoitustyön, joka arvostellaan. Luvussa 4 esitel-

lyistä arviointimenetelmistä mikään ei tarjoa realistista vaihtoehtoa kurssin nykyiselle arviointimenetelmälle.

Opettajan kannalta kurssi on kuitenkin lähes suuritöisin siihen liittyvän ohjaus- ja arviointityön kannalta. Kurssin arviointivastuuta voisi kuitenkin jonkin verran hajauttaa opiskelijoille. Harjoitustyön dokumenttiin olisi hyvä liittää lyhyt itsearviointiosuus, jossa opiskelija arvioi oman työnsä käyttäen samaa arviointikriteeristöä kuin opettaja ja ehdottaa itselleen arvosanaa. Arviointia saadaan selkeytettyä ottamalla kurssilla käyttöön nykyisen arvioinnin sijaan luvussa 4.3 esitetty matriisiarviointi.

Itsearviointiin lisäksi kurssin suorittamiseksi voisi kokeilla myös vertaisarviointia niin, että jokainen kurssille osallistuja arvioi toisen opiskelijan työn. Vertaisarviointia varten arvioitavien töiden aiheiden tulisi olla kutakuinkin samantasoisia. Vertaisarviointiin kannattaa käyttää samoja kurssin arviointiin laadittuja yleisiä arviointikriteerejä.

Harjoitustyön laajuudesta johtuen vertaisarviointiin ja itsearviointiin onnistuminen riippuu pitkälti arvioinnin suorittavista opiskelijoista. Ihannetapauksessa opiskelijoiden olisi syytä arvioida useampia töitä harjaantuakseen arviointiin. Kurssi on kuitenkin niin suuritöinen, että varsinaisen harjoitustyön lisäksi kurssin opintopistemäärään ei mahdu kovin paljoa lisätyötä. Mikäli itsearviointia ja vertaisarviointia päädytään käyttämään kurssilla, tulee opettajan kuitenkin varata oikeus määrittää lopullinen arvosana. Vertaisarviointiin ja itsearviointiin onnistumista näin laajan ja perehtymistä vaativan kurssin puitteissa on vaikea arvioida etukäteen, ennen kun niitä on testattu käytännössä aidolla kurssilla.

6.4.6 Diskreetit rakenteet

Diskreetit rakenteet kurssi poikkeaa muista ViSCoS-kursseista siten, että opiskeltava tieto on pääosin matematiikkaa. Kurssilla käytetään tällä hetkellä arviointimenetelmänä perinteistä tenttiä. Kurssin sisällön vuoksi perinteinen tentti sopii tälle kurssille paremmin kuin muille ViSCoS-kursseille. Tenttiin liittyvää kaksoisrakenteellisuutta voidaan entisestään vähentää laatimalla tehtävät todellisia sovellustilanteita vastaaviksi. Kurssilla opiskelijat tekevät harjoituksia, joilla ei nyt ole merkittävää vaikutusta kurssin arviointiin. Harjoitustehtävät voi kuitenkin tälläkin kurssilla ottaa laajemmin mukaan arvosteluun, esimerkiksi siten, että harjoitustehtävistä saatava arviointi muodostaa puolet kurssin arvostelusta.

Matemaattisen tiedon soveltamistaitoja voidaan arvioida ViSCoS-kurssilla esimerkiksi verkkotenteillä tai kotitenteillä. Myös hyvin laaditut monivalintakokeet soveltuvat matemaattisen tiedon arviointiin. Koska hyvien monivalintatehtävien suunnittelu vaatii paljon opettajan resursseja, niiden käyttäminen kurssilla ei ole järkevää. Verkkotenttien käyttämisessä ongelmaksi saattavat muodostua matemaattisten symbolien käyttäminen verkko-oppimisympäristössä.

Tässä suhteessa kotitenttien käyttäminen kurssilla on parempi vaihtoehto. Kotitenttien käyttäminen arvioinnissa poistaa paperisten tenttien postittamiseen liittyvät ongelmat. Opiskelijan ei tarvitse tenttiin osallistuakseen matkustaa tenttipaikkaan. Esitetty harjoitustehtävät ja kotitentit yhdistävä arviointi vastaa myös hyvän yliopistollisen arviointimenetelmän vaatimuksiin. Kotitenttiin liittyy kohtalaisen suuri vilpin riski. Vilpin esiintymisen todennäköisyyttä voidaan kuitenkin vähentää käyttämällä useampia erilaisia tenttejä, sekä liittämällä osaksi tenttiä itsearviointiosuus, jossa opiskelijat arvioivat omaa tenttisuoritustaan sekä oppimistaan kurssilla.

6.4.7 Tietokoneen rakenne ja käyttöjärjestelmä

Tietokoneen rakenne ja käyttöjärjestelmä -kurssin nykyisen arvioinnin ongelmana on kaksoisrakenteisuus. Kurssilla opiskeltava asia on melko epäyhtenäistä ja se koostuu useammasta osa-alueesta. Myös tällä kurssilla harjoitustehtävät pitäisi saada paremmin mukaan kurssin arviointiin.

Kurssilla opiskeltavan asian kannalta harjoitustehtävät ovat tärkeitä etenkin lukujärjestelmämuunnosten sekä tietokoneen läheisen ohjelmoinnin oppimisen kannalta. Kurssin asiasisällön ja rakenteen kannalta kurssille soveltuu samankaltainen arviointi, kuin Tietotekniikan perusteet -kurssille (luku 6.4.1). Koska harjoitustehtävien arviointia tulisi kurssilla painottaa, harjoitustehtävien osuus kurssin kokonaisarvostelussa voisi olla esimerkiksi 75 % ja loput 25 % arvioinnista voisi määräytyä jostain kurssilla käsitellyä asiaa laajemmin käsittelevän kirjoitelman perusteella.

6.4.8 Johdatus tietojenkäsittelyn etiikkaan

Johdatus tietojenkäsittelyn etiikkaan -kurssin arviointi vastaa nyt hyvin yliopistollisen arvioinnin periaatteita, eikä arviointiin liity luvun 6.2.8 tarkastelun valossa erityisiä

ongelmia. Botting (2005) on vastaavaa kurssia opettaessaan käyttänyt opiskelijoiden arviointiin arvosanoja. Kirjoittajan mukaan arvosanojen antaminen kirjoitelmista lisää opiskelijoiden motivaatiota. Nykyinen arviointitapa toimii kuitenkin ViSCoS-kurssilla paremmin, sillä etiikkaa käsittelevän kirjoitelman ja oppimispäiväkirjan arvosteleminen numeroarvosanoin objektiivisesti on hankalaa. Kurssin tavoitteena on aktivoida opiskelijoita ajattelemaan millaisia eettisiä ongelmia tietojenkäsittelyn alaan liittyy, eikä arviointi arvosanoin ole tässäkään suhteessa perusteltua.

6.5 Uudet arviointimenetelmät ja palaute ViSCoS-opinnoissa

Luvussa 6.4 esitin kuinka ViSCoS-kurssien arviointia voitaisiin kehittää. Taulukkoon 3 olen laatinut tiiviin yhteenvedon, siitä minkälaiset arviointimenetelmät ovat kursseilla käytössä nykyisin ja miten suosittelen arviointia kehitettävän. Taulukosta ilmenee myös suosittamani ratkaisu arvioinnin painottamisesta kursseilla.

Kuten aikaisemmin totesin, koko esitetty arviointiratkaisu perustuu kokemukseeni kursseista ja niiden opettamisesta, kurssien sisällöstä sekä kirjallisuudessa olevista suosituksista tietojenkäsittelytieteen kurssien arviointiin. Kurssien lopullisesta arvioinnista vastaa kuitenkin kurssista vastaava opettaja. Erityisesti eri arviointimenetelmien painotukset riippuvat siitä, mitä asioita halutaan korostaa tärkeinä arviointia ja arvostelua tehtäessä.

Uusia arviointimenetelmiä käyttöönotettaessa on tärkeää tiedostaa se, että jokaisella arviointikerralla arviointitavasta ja sen käytöstä opitaan lisää. Arviointimenetelmiin tottuminen vie aikaa sekä opettajalta että opiskelijalta (Karjalainen ja Kemppainen, 1994). Tässä suhteessa pitkälle meneviä johtopäätöksiä arviointimenetelmien soveltuvuudesta kursseille saadaan vasta, kun niitä on käytetty kurssilla muutamia kertoja ja muokattu kurssin tarpeita vastaavaksi.

Otettaessa käyttöön näitä arviointimenetelmiä tulee varautua myö siihen, että opettajan työmäärä kursseilla kasvaa kertaluontoisesti. Uudet arviointimenetelmät pitää ohjeistaa ja niitä varten pitää laatia jonkin verran tukimateriaalia, kuten esseekysymyksiä. Moodle tarjoaa jonkin verran valmiita työkaluja arvioinnin käytännön toteutukseen. Tämän lisäksi on saatavilla useita käyttäjien laatimia Moodlen toiminnallisuutta laajentavia työkaluja, joita voidaan käyttää hyödyksi. En kuitenkaan tarkastele näitä työkaluja ja niiden käyttöä tämän työn puitteissa. Työkaluihin on kuitenkin syytä tutustua

Taulukko 3: Yhteenveto suositelluista arviointimenetelmistä ViSCoS-kursseille

Kurssi	Nykyinen arviointi	Suosittelut arviointi ja painotukset
Johdatus tietotekniikkaan	harjoitustehtävät perinteinen tentti HTML-harjoitus	harjoitustehtävät 50 % essee 50 % HTML-harjoitus (hyv/hyl)
Ohjelmointi 1	harjoitustehtävät perinteinen tentti	harjoitustehtävät 60 % kokoava harjoitustehtävä 40 % oppimispäiväkirja (hyv/hyl)
Tietojenkäsittelytieteen tutkimusaloja	ryhmätyökirjoitelma oppimispäiväkirja	ryhmätyökirjoitelma (hyv/hyl) oppimispäiväkirja (hyv/hyl)
Ohjelmointi 2	harjoitustehtävät perinteinen tentti	harjoitustehtävät 60 % 1. kokoava harjoitustehtävä 20 % 2. kokoava harjoitustehtävä 20 % oppimispäiväkirja (hyv/hyl)
Perusopintojen harjoitustyö	harjoitustyön pisteytykseen perustuva arviointi	matriisiarviointiin perustuva harjoitustyön arviointi sekä itsearviointi ja vertaisarviointi
Diskreetit rakenteet	harjoitustehtävät perinteinen tentti	harjoitustehtävät 50 % kotitentti 50 %
Tietokoneen rakenne ja käyttöjärjestelmä	harjoitustehtävät perinteinen tentti	harjoitustehtävät 75 % essee 25 %
Johdatus tietojenkäsittelyn etiikkaan	essee oppimispäiväkirja	essee (hyv/hyl) oppimispäiväkirja (hyv/hyl)

uudistettuja kursseja laadittaessa, sillä niillä voidaan helpottaa esimerkiksi vertaisarvioinnin käytännön toteutusta.

Edellä mainittujen kurssikohtaisten arvioinnin parantamiseen tähtäävien toimenpiteiden lisäksi on syytä kehittää palautteen keräämistä ja antamista koko ViSCoS-opinnoissa. Repo-Kaarento (1999) pitää opettajan ja opiskelijoiden välisen molemminpuolisen palautteen antamista tärkeänä kurssin kehittämisen ja opiskeli-

jan oppimisen kannalta. Kirjoittajan mukaan avoin kaksisuuntainen palauteympäristö mahdollistaa sen, että opettaja ja opiskelija oppivat toisiltaan. Pölonen (2005) osoittaa, että nykyinen ViSCoS-kursseilla saatava palaute ei riitä kaikille opiskelijoille. Ongelma koskee erityisesti ohjelmointikursseja.

Tällä hetkellä ViSCoS-kurssien opettajat ovat keränneet kurssipalautetta harjoitustehtävien muodossa. Jokaisella kurssilla on kuitenkin hieman erilainen tapa kerätä palautetta, eikä palautetietoa välttämättä käytetä kovin tehokkaasti hyödyksi kursseja kehitettäessä. TieVie – Tieto- ja viestintätekniiikan opetuskäytön virtuaaliyliopistohanke (2002) on kerännyt verkkosistolle useita esimerkkejä opiskelijoilta kerättävän palautteen kehittämiseen sekä käyttämiseen. Sivustolla on myös käsitelty palautteen antamista opiskelijalle. Verkkosivuston esimerkkejä olisi syytä hyödyntää myös ViSCoS-opinnoissa.

ViSCoS-opintojen kannalta olisi järkevää laatia yhdenmukainen palautejärjestelmä. Palautteen kerääminen on tärkeätä myös ViSCoS-opintoihin liittyvän tutkimustyön kannalta, joten palautteen kokoaminen tulisi järjestää niin, että palautetieto on tuotavissa helposti Moodlesta tai verkkosivulta, jossa palautetta kerätään. Myös palautteen antamisen ja pyytämisen kynnyksiä tulisi madaltaa koko ViSCoS-opintojen osalta.

7 Yhteenveto

ViSCoS-opinnot ovat 25 ECTS-opintopisteen laajuiset tietojenkäsittelytieteen perusopinnot, jotka opetetaan Internetin välityksellä. Opinnot tarjotaan nykyisin suoritettavaksi Joensuun yliopiston täydennyskoulutuskeskuksen Avoimen yliopiston kautta ja niitä voi suorittaa myös englanniksi ulkomailta käsin. Opintojen arvioinnin kehittäminen on tarpeellista, sillä useimmilla kursseilla pääasiallisena arviointimenetelmänä ovat perinteiset tentit, joihin liittyy useita ongelmia. Tenttien laatiminen on opettajien näkökulmasta työlästä ja tenttitilaisuuksien järjesteleminen on hankalaa. Tenttien postitusajat sekä aikarajoitukset tenttien arvostelussa aiheuttavat sen, että ViSCoS-kurssien opettajien työpanos kuluu ajoittain pelkästään tenttien arviointiin ja tentteihin liittyviin käytännön järjestelyihin.

Opiskelijan arviointiin on kehitetty useita erilaisia menetelmiä ja lisää kehitetään jatkuvasti. Arviointiin käytettävät ja valittavat menetelmät riippuvat opetettavasta asiasta, sekä siitä, millaista tietoa opiskelijan osaamisesta halutaan nostaa esille. Arviointitieto välitetään yleensä opiskelijalle joko arviointina tai palautteena, mutta arviointitieto voi myös jäädä pelkästään opettajan käyttöön. Opiskelijan arviointiin käytettävä menetelmä määrittää pitkälti sen, mitä opiskelija muistaa opetuksesta, joten arviointimenetelmän valinnalla on suuri merkitys opiskelijan oppimisen kannalta.

Hyvä yliopisto-opintojen arviointimenetelmä opettaa opiskelijaa etsimään, löytämään, arvioimaan ja soveltamaan tietoa. Hyvän arviointimenetelmän ominaisuuksiin kuuluu myös, että se valmentaa opiskelijaa toimimaan työelämässä opettamalla vuorovaikutus- ja arviointitaitoja. Perinteiseen tenttiin liittyy kaksoisrakenteen ongelma, sillä perinteinen tentti arviointimenetelmänä toimii enemmän ulkomuistin kuin oppimisen mittarina. Arviointimenetelmiä valittaessa niihin liittyvä kaksoisrakenne pitää tunnistaa ja pyrkiä minimoimaan.

Tietojenkäsittelytieteen opintojen arvioinnissa keskitytään yleensä opiskelijoiden ongelmanratkaisukyvyyn arviointiin. Oman haasteensa tietojenkäsittelytieteen opintojen arvioinnille antaa ohjelmoinnin osaamisen arviointi. Perinteiset tentit soveltuvat erittäin huonosti ohjelmointitaitojen arvioimiseen, sillä kynä ja paperi eivät ole opiskelijoille tuttuja työvälineitä ohjelmointityössä, eikä tällaisella kynällä ”ohjelmoinnilla” ole vastinetta työelämässä. Opiskelijan arviointiin on tietojenkäsittelytieteen kursseilla usein käytetty tietokoneavusteisia arviointivälineitä välittömän palautteen ja arvostelun

antamiseksi. Näistä menetelmistä ViSCoS-opintojen kannalta mielenkiintoisimpia ovat automaattinen essee-tehtävien arviointi sekä automaattinen ja puoliautomaattinen ohjelmointitehtävien arviointi. Valitettavasti kumpikaan ei nykymuodossaan sovellu käytettäväksi ViSCoS-opintojen arvioinnissa.

Tämän tutkielman ensimmäisenä tutkimuskysymyksenä oli kartoittaa millaisia tietojenkäsittelytieteen verkko-opintoihin soveltuvia arviointimenetelmiä on olemassa. Luvussa 4 olen esitellyt joukon yleisesti käytettyjä arviointimenetelmiä, joita voitaisiin hyödyntää myös ViSCoS-kursseilla. Jokainen esitelty arviointimenetelmä antaa erilaista tietoa opiskelijan osaamisesta. Arviointimenetelmät ovat yksilöllisiä, joten soveltuvan arviointimenetelmän valinta riippuu siitä, mitä ollaan arvioimassa. Erilaisia arviointimenetelmiä on kuitenkin olemassa lukematon määrä ja ihannetapauksessa arviointimenetelmä räätälöidään aina kurssikohtaisesti.

Toisena tutkimuskysymyksenä oli selvittää millaisia ominaisuuksia ViSCoS-opintojen arviointiin käytettäviltä menetelmiltä vaaditaan. ViSCoS-opinnot ovat korkeakouluopintoja, joten hyvän yliopistollisen arviointimenetelmän kriteerit ovat luontevasti osa ViSCoS-opintojen arviointiin käytettävien menetelmien kriteereitä. Muita menetelmältä vaadittuja ominaisuuksia on, että se antaa mahdollisimman tarkan kuvan opiskelijan osaamisesta, arviointi on opiskelijan näkökulmasta positiivinen kokemus ja arviointi on avointa. Koska vilpin esiintymiseen tulee varautua, on tärkeää, että käytettävässä arviointimenetelmässä vilpin tekeminen on mahdollisimman epätodennäköistä ja vilppi on tunnistettavissa kohtuullisella vaivalla. Lisäksi on tärkeitä, että arviointi on mahdollisimman nopeaa ja helppoa opiskelijalle sekä opettajalle. Arvioinnin tulee myös olla ajasta ja paikasta riippumatonta. Valitettavasti kaikki nämä kriteerit hyvin täyttävää menetelmää on mahdotonta löytää.

ViSCoS-opintojen näkökulmasta arviointia on luontevinta lähteä kehittämään kurssikohtaisesti. Suurin osa ViSCoS-kursseista arvioidaan nykyisin perinteisten tenttien avulla ja tämä tuo esiin kaksoisrakenteen ongelman. Erityisen voimakkaana kaksoisrakenteisuus näkyy ohjelmointikursseilla. Myös muiden kurssien nykyiseen arviointiin liittyy ongelmia. ViSCoS-kurssien arvioinnin kannalta ei ole mielekäästä säilyttää nykyisiä arviointitilaisuuksia, vaan arviointi tulisi järjestää Internetin kautta. Tämä johtaa vilpillisen toiminnan riskin kasvamiseen, mutta kuten luvussa 6.3 totean, tämä riski on syytä ottaa ja tiedostaa.

Viimeisenä tutkimuskysymyksenä oli tarkastella sitä, mitä arviointimenetelmiä ViSCoS-opintojen ja -kurssien arvioinnissa tulisi tulevaisuudessa hyödyntää. Olen keskittynyt uudistettujen kurssikohtaisten arviointimenetelmien valinnassa kaksoisrakenteen poistamiseen arvioinnista sekä harjoitustehtävien liittämiseen osaksi arviointia. Erityisen tärkeitä nämä muutokset ovat Ohjelmointi 1 ja Ohjelmointi 2 -kurssien osalta, joiden nykyiseen arviointiin liittyy eniten kaksoisrakenteisuutta. Tietojenkäsittelytieteen tutkimusaloja ja Johdatus tietojenkäsittelyn etiikkaan -kurssien arviointi vastaa jo nykyisessä muodossaan kohtuullisen hyvin vaatimuksia hyvälle ViSCoS-kurssin arvioinnille.

Uudistetuiksi arviointimenetelmiksi Ohjelmointi 1 ja Ohjelmointi 2 -kursseille suosittelen harjoitustehtävien ottamista suuremmaksi osaksi arviointia, kokoavien harjoitustehtävien teettämistä sekä oppimispäiväkirjaa. Johdatus tietotekniikkaan sekä Tietokoneen rakenne ja käyttöjärjestelmä -kursseille esitän perinteisen tentin sijasta kirjoitelmiin ja harjoitustehtäviin perustuvaa arviointia. Perusopintojen harjoitustyö -kursseille suosittelen itsearvioinnin ja vertaisarvioinnin lisäämistä kurssille opettajan arviointityön helpottamiseksi. Matemaattispainotteiselle Diskreetit rakenteet -kursseille esitän vaihtoehtoisena arviointitapana kotitenttiä ja harjoitustehtävien laajempaa hyödyntämistä arvioinnissa. Kaikille ViSCoS-kursseille yhtenäisen kaksisuuntaisen palautejärjestelmän kehittäminen on myös tärkeää arvioinnin parantamiseksi.

Tämän tutkielman perusteella ViSCoS-kurssien arvioinnin kehittäminen on tärkeää. Suurimmalla osalla kursseista esiintyy nykyisin kaksoisrakenteisuutta, joka ei kuulu hyvään yliopistolliseen arviointiin ja on haitallista sekä opiskelijan että opettajan näkökulmasta. Suosittelenkin, että ViSCoS-kurssien arviointia muutetaan kaksoisrakenteisuuden poistamiseksi ja harjoitustehtävien arvostuksen lisäämiseksi. Vähimmäisvaatimuksena arvioinnin kehittämiseksi on muuttaa ohjelmointikurssien arviointia.

Viitteet

- Karen Anewalt. Experiences Teaching Writing in a Computer Science Course for the First Time. *Journal of Computing Sciences in Colleges*, **18**, no. 2, 346–355, 2002.
- J. Bari, R. Sullivan ja J. Blair. Security Method in Distance-Learning. Teoksessa *Frontiers in Education, 2004. FIE 2004. 34th Annual*. 2004.
- Mordechai Ben-Ari. Constructivism in Computer Science Education. *Journal of Computers in Mathematics and Science Teaching*, **20**, no. 1, 45–73, 2001.
- Blackboard Inc. *Blackboard verkko-oppimisympäristön WWW-sivu*. <http://www.blackboard.com/> Tarkistettu 10.11.2007.
- Richard J. Botting. Teaching and Learning Ethics in Computer Science: Walking the Walk. Teoksessa *SIGCSE '05: Proceedings of the 36th SIGCSE technical symposium on Computer science education*, sivut 342–346, New York, NY, USA. ACM, 2005. ISBN 1-58113-997-7.
- Philip J. Burton ja Russel E. Bruhn. Teaching Programming in the OOP Era. *ACM SIGCSE Bulletin*, **35**, no. 2, 111–114, 2003.
- Kevin Bushweller. Generation of Cheaters. *The American School Board Journal*, **186**, no. 4, 24–32, 1999.
- Janet Carter, Kirsti Ala-Mutka, Ursula Fuller, Martin Dick, John English, William Fone ja Judy Sheard. How Shall We Assess This? Teoksessa *ITiCSE-WGR '03: Working group reports from ITiCSE on Innovation and technology in computer science education*, sivut 107–123, New York, NY, USA. ACM Press, 2003.
- Committee on the Foundations of Assessment, National Research Council. *Knowing What Students Know: The Science and Design of Educational Assessment*. National Academy press Washington, D.C, 2001.
- Jose L. Cordova ja Paula Thornhill. Academic Honesty and Electronic Assessment: Tools to Prevent Students From Cheating Online—Tutorial Presentation. *Journal of Computing Sciences in Colleges*, **22**, no. 5, 52–54, 2007.
- Charlie Daly. RoboProf and an Introductory Computer Programming Course. Teoksessa *ITiCSE '99: Proceedings of the 4th annual SIGCSE/SIGCUE ITiCSE conference*

on Innovation and technology in computer science education, sivut 155–158, New York, NY, USA. ACM, 1999. ISBN 1-58113-087-2.

Charlie Daly ja John Waldron. Assessing the Assessment of Programming Ability. Teoksessa *SIGCSE '04: Proceedings of the 35th SIGCSE technical symposium on Computer science education*, sivut 210–213, New York, NY, USA. ACM, 2004. ISBN 1-58113-798-2.

Martin Dick, Judy Sheard, Cathy Bareiss, Janet Carter, Donald Joyce, Trevor Harding ja Cary Laxer. Addressing Student Cheating: Definitions and Solutions. Teoksessa *ITiCSE-WGR '02: Working group reports from ITiCSE on Innovation and technology in computer science education*, sivut 172–184, New York, NY, USA. ACM, 2002.

Christopher Douce, David Livingstone ja James Orwell. Automatic Test-based Assessment of Programming: A Review. *ACM Journal of Educational Resources in Computing*, **5**, no. 3, 4, 2005a.

Christopher Douce, David Livingstone, James Orwell, Steve Grindle ja Justin Cobb. A Technical Perspective on ASAP — Automated System for Assessment of Programming. Teoksessa *Proceedings of the 9th CAA Conference, Loughborough*. Loughborough University, 2005b.

S. Fincher. What Are We Doing When We Teach Programming? Teoksessa *Frontiers in Education Conference, 1999. FIE '99. 29th Annual*, osa 1, sivut 12A4/1–12A4/5 vol.1. 1999.

R. Steven Flynn. Confronting Academic Dishonesty in the Accounting Classroom: A Cultural Experience? *Accounting Education*, **12**, no. 4, 437–439, 2003.

Susan E. George. Learning and the Reflective Journal in Computer Science. Teoksessa *ACSC '02: Proceedings of the twenty-fifth Australasian conference on Computer science*, sivut 77–86, Darlinghurst, Australia. Australian Computer Society, Inc., 2002.

Hasmik Gharibyan. Assessing Students' Knowledge: Oral Exams vs. Written Tests. Teoksessa *ITiCSE '05: Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education*, sivut 143–147, New York, NY, USA. ACM Press, 2005. ISBN 1-59593-024-8.

- Graham Gibbs ja Claire Simpson. Conditions Under Which Assessment Supports Students' Learning. *Learning and Teaching in Higher Education*, **1**, 3–31, 2004.
- Arto Haataja, Jarkko Suhonen, Erkki Sutinen ja Sirpa Torvinen. High School Students Learning Computer Science Over the Web. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, **3**, no. 2, 2001. Saatavissa: <http://imej.wfu.edu/articles/2001/2/04/index.asp>.
- Colin Higgins, Tarek Hegazy, Pavlos Symeonidis ja Athanasios Tsintsifas. The CourseMarker CBA System: Improvements over Ceilidh. *Education and Information Technologies*, **8**, no. 3, 287–304, 2003.
- Mary Hricko, toimittaja. *Online Assessment and Measurement: Foundations and Challenges*. Hershey, PA, USA, Information Science Publishing, 2005.
- Olli Hyppönen. *Erilaisia oppimisen arviointimenetelmiä - Kuvaukset, vahvuudet ja haasteet*. TKK Opetuksen ja opiskelun tuki, <http://www.dipoli.tkk.fi/ok/p/menetelmat/oppimisenarviointimenetelmat.pdf> Tarkistettu 22.4.2008.
- David Jackson. A Semi-Automated Approach to Online Assessment. *ACM SIGCSE Bulletin*, **32**, no. 3, 164–167, 2000.
- David Jackson ja Michelle Usher. Grading Student Programs Using ASSYST. *ACM SIGCSE Bulletin*, **29**, no. 1, 335–339, 1997.
- Joensuun yliopisto. *Opintosuorituksia ja opiskelijan oikeusturvaa koskeva johtosääntö*. <http://www.joensuu.fi/mltdk/oppaat/oikeusturva.pdf> Tarkistettu 18.10.2007.
- Mike Joy, Nathan Griffiths ja Russell Boyatt. The Boss Online Submission and Assessment System. *Journal on Educational Resources in Computing (JERIC)*, **5**, no. 3, 2, 2005.
- Mike Joy ja Michael Luck. Effective Electronic Marking for On-Line Assessment. Teoksessa *ITiCSE '98: Proceedings of the 6th annual conference on the teaching of computing and the 3rd annual conference on Integrating technology into computer science education*, sivut 134–138, New York, NY, USA. ACM, 1998. ISBN 1-58113-000-7.

- T. Kakkonen ja E. Sutinen. Automatic Assessment of the Content of Essays Based on Course Materials. Teoksessa *2nd International Conference on Information Technology: Research and Education, 2004.*, sivut 126–130. 2004.
- Osku Kannusmäki, Andrés Moreno, Niko Myller ja Erkki Sutinen. What a Novice Wants: Students Using Program Visualization in Distance Programming Course. Teoksessa *Proceedings of the Third Program Visualization Workshop (toim. A. Korhonen), Research Report CS-RR-407*, sivut 126–133. Department of Computer Science, University of Warwick, UK, 2004. ISBN 0-902683-74-8.
- Anja Kareinen, Niko Myller, Jarkko Suhonen ja Erkki Sutinen. Visuals First Approach in an Introductory Web-based Programming Course. Teoksessa Mordechai Ben-Ari, toimittaja, *Proceedings of the Second Program Visualization Workshop, HornstrupCentret, Denmark*. Department of Computer Science, University of Aarhus., 2002.
- Asko Karjalainen. *Tentin teoria*. Väitöskirja, Oulun yliopisto, kasvatustieteiden tiedekunta, 2001.
- Asko Karjalainen ja Tiina Kemppainen. *Vaihtoehtoisia tenttikäytäntöjä – ohjeita ja ideoita yliopistotenttien kehittämiseen*. Oulun yliopiston opintotoimisto, 1994.
- Taina Lehtonen. *Automaattinen ohjelmakoodin tarkastus – Boss- ja CourseMaster-järjestelmien soveltuvuus ohjelmoinnin virtuaaliopintoihin*. Pro Gradu, Joensuun yliopisto, tietojenkäsittelytieteen laitos, 2002.
- Timo Lehtonen. Javala — Addictive E-Learning of the Java Programming Language. Teoksessa Tapio Salakoski, Tomi Mäntylä ja Mikko Laakso, toimittajat, *Proceedings of Koli Calling 2005 – Fifth Koli Calling Conference on Computer Science Education*, TUCS General Publication No 41, sivut 41–48. Turku Centre for Computer Science, 2006.
- Alberta Lipson ja Norma McGavern. Undergraduate Academic Dishonesty at MIT. Results of a Study of Attitudes and Behavior of Undergraduates, Faculty, and Graduate Teaching Assistants. 1993.
- Raymond Lister. Objectives and Objective Assessment in CS1. *ACM SIGCSE Bulletin*, **33**, no. 1, 292–296, 2001.

- Romans Lukashenko, Vita Graudina ja Janis Grundspenkis. Computer-Based Plagiarism Detection Methods and Tools: An Overview. Teoksessa *CompSysTech '07: Proceedings of the 2007 international conference on Computer systems and technologies*, sivut 1–6, New York, NY, USA. ACM, 2007. ISBN 978-954-9641-50-9.
- Jaana Lukkarinen. *Suomalaisista verkko-oppimisympäristöistä*. Pro Gradu, Joensuun yliopisto, tietojenkäsittelytieteen laitos, 2003.
- Lauri Malmi, Ari Korhonen ja Riku Saikkonen. Experiences in Automatic Assessment on Mass Courses and Issues for Designing Virtual Courses. Teoksessa *ITiCSE '02: Proceedings of the 7th annual conference on Innovation and technology in computer science education*, sivut 55–59, New York, NY, USA. ACM, 2002. ISBN 1-58113-499-1.
- Jerry Mead, Simon Gray, John Hamer, Richard James, Juha Sorva, Caroline St. Clair ja Lynda Thomas. A Cognitive Approach to Identifying Measurable Milestones for Programming Skill Acquisition. Teoksessa *ITiCSE-WGR '06: Working group reports on ITiCSE on Innovation and technology in computer science education*, sivut 182–194, New York, NY, USA. ACM, 2006.
- Veijo Meisalo, Erkki Sutinen ja Jorma Tarhio. *Modernit oppimisympäristöt*. Tietosanomaa Oy, 2003.
- Ulf Melin ja Stefan Cronholm. Project Oriented Student Work: Learning & Examination. Teoksessa *ITiCSE '04: Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education*, sivut 87–91, New York, NY, USA. ACM Press, 2004. ISBN 1-58113-836-9.
- Moodle-yhteisö. *Moodle oppimisympäristön WWW-sivu*. <http://moodle.org> Tarkistettu 10.11.2007.
- Barbara Moskal, Keith Miller ja L. A. Smith King. Grading Essays in Computer Ethics: Rubrics Considered Helpful. Teoksessa *SIGCSE '02: Proceedings of the 33rd SIGCSE technical symposium on Computer science education*, sivut 101–105, New York, NY, USA. ACM, 2002. ISBN 1-58113-473-8.
- Pirkko Pölönen. *Ohjelmoinnin verkkokurssien ohjaaminen – Case ViSCoS*. Pro Gradu, Joensuun yliopisto, tietojenkäsittelytieteen laitos, 2005.

- Saara Repo-Kaarento. *Opetus sydämen asiana*, luku Palaute on lahja, sivut 145 – 148. Dialogeja 2. Oulun yliopisto, 1999.
- Anthony Robins, Janet Rountree ja Nathan Rountree. Learning and Teaching Programming: A Review and Discussion. *Computer Science Education*, **13**, no. 2, 137–172, 2003.
- N.C. Rowe. Cheating in online student assessment: Beyond plagiarism. *Online Journal of Distance Learning Administration*, **7**, no. 2, 2004. Saatavissa: <http://www.westga.edu/~distance/ojdla/summer72/rowe72.html>.
- Riku Saikkonen, Lauri Malmi ja Ari Korhonen. Fully automatic assessment of programming exercises. Teoksessa *ITiCSE '01: Proceedings of the 6th annual conference on Innovation and technology in computer science education*, sivut 133–136, New York, NY, USA. ACM, 2001. ISBN 1-58113-330-8.
- Sidney L. Sanders. Computer Science Homework and Grading Practices: An Alternative to the Popular Model. *ACM SIGCSE Bulletin*, **20**, no. 2, 26–29, 1988.
- William L. Sanders ja Sandra P. Horn. Educational Assessment Reassessed: The Usefulness of Standardized and Alternative Measures of Student Achievement as Indicators for the Assessment of Educational Outcomes. *Education Policy Analysis Archives*, **13**, no. 30, 1995.
- Judy Sheard, Angela Carbone ja Martin Dick. Determination of Factors Which Impact on IT Students' Propensity to Cheat. Teoksessa *ACE '03: Proceedings of the fifth Australasian conference on Computing education*, sivut 119–126, Darlinghurst, Australia, Australia. Australian Computer Society, Inc., 2003. ISBN 0-909925-98-4.
- Judy Sheard ja Martin Dick. Influences on Cheating Practice of Graduate Students in IT Courses: What Are the Factors? Teoksessa *ITiCSE '03: Proceedings of the 8th annual conference on Innovation and technology in computer science education*, sivut 45–49, New York, NY, USA. ACM Press, 2003. ISBN 1-58113-672-2.
- B. A. Sheil. The Psychological Study of Programming. *ACM Computing Surveys*, **13**, no. 1, 101–120, 1981.
- Simon. Electronic Watermarks to Help Authenticate Soft-copy Exams. Teoksessa *ACE '05: Proceedings of the 7th Australasian conference on Computing education*, sivut

7–13, Darlinghurst, Australia, Australia. Australian Computer Society, Inc., 2005. ISBN 1-920682-24-4.

Jirarat Sitthiworachart ja Mike Joy. Effective Peer Assessment for Learning Computer Programming. Teoksessa *ITiCSE '04: Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education*, sivut 122–126, New York, NY, USA. ACM, 2004. ISBN 1-58113-836-9.

Erkki Sutinen ja Sirpa Torvinen. The Candle Scheme for Creating an on-line Computer Science Program - Experiences and Vision. *Informatics in Education*, **2**, 93–102, 2003.

The Joint Task Force on Computing Curricula. *Computing Curricula 2001*. <http://www.sigcse.org/cc2001/> Tarkistettu 23.10.2007.

Pete Thomas, Linda Carswell, Blaine Price ja Marian Petre. A Holistic Approach to Supporting Distance Learning Using the Internet: Transformation, Not Translation. *British Journal of Educational Technology*, **29**, no. 2, p149, 1998.

Pete Thomas, Blaine Price, Marian Petre, Linda Carswell ja Mike Richards. Experiments with Electronic Examinations over the Internet. Teoksessa *Fifth International Computer Assisted Assessment Conference, Loughborough University, Loughborough, UK*. 2001.

Tietojenkäsittelytieteen opetusta järjestävät yliopistot. *Tietojenkäsittelytieteen yhteisvalinnan aikaisemmat pääsykokeet*. <http://www.tkt-yhteisvalinta.fi/?page=Aiemmat%20kokeet> Tarkistettu 30.1.2008.

TieVie – Tieto- ja viestintätekniiikan opetuskäytön virtuaaliyliopistohanke. *Arviointiaavan sanomat 1/2002: Opetuksen ja oppimisen arviointi*. http://tievie.oulu.fi/arviointiaavan_sanomat/ Tarkistettu 6.1.2008.

Sirpa Torvinen. *Ohjelmoinnin perusteiden oppimisen ongelmia virtuaalisessa oppimisympäristössä*. Pro Gradu, Joensuun Yliopisto, tietojenkäsittelytieteen laitos, 2001.

Sirpa Torvinen. *Aspects of the Evaluation and Improvement Process in an Online Programming Course, Case: the ViSCoS program*. Lisensiaattityö, Joensuun Yliopisto, 2004.

UIAH Media Lab, Taideteollinen korkeakoulu. *Fle3 > Future Learning Environment WWW-sivu*. <http://fle3.uiah.fi/> Tarkistettu 10.11.2007.

- Gary Ury ja Pat McFarland. Authentic Assessment Through Electronic Portfolios. Teoksessa *Proceedings of the seventh annual consortium for computing in small colleges central plains conference on The journal of computing in small colleges*, sivut 353–368, , USA. Consortium for Computing Sciences in Colleges, 2001.
- Salvatore Valentia, Francesca Neri ja Alessandro Cucchiarelli. An Overview of Current Research on Automated Essay Grading. *The Journal of Information Technology Education*, **2**, 319–330, 2003.
- Ben Wilbrink. Assessment in Historical Perspective. *Studies in Educational Evaluation*, **23**, 31–48(18), 1997. Saatavissa: <http://www.benwilbrink.nl/publicaties/97AssessmentStEE.htm>.
- Karyn Woodford ja Peter Bancroft. Multiple Choice Questions Not Considered Harmful. Teoksessa *ACE '05: Proceedings of the 7th Australasian conference on Computing education*, sivut 109–116, Darlinghurst, Australia. Australian Computer Society, Inc., 2005. ISBN 1-920682-24-4.

Liite 1: Esimerkki matriisiarvioinnista

TieVie – Tieto- ja viestintätekniikan opetuskäytön virtuaaliyliopistohanke (2002) on julkaissut seuraavan esimerkin matriisiarviointiin käytettävän matriisista.

OPETUKSEN LAADUN ITSEARVIOINTI LIITE 2 LAITOSTEN ITSEARVIOINTIEN RUBRIIKKI	TOIMINTA ON RUTIINIMAISTA EIKÄ SITÄ TUETA KEHITTÄMISTOIMIN	KEHITTÄMISTOIMINTAA ON JONKIN VERRAN, LAITOKSEN TUKITOIMET HEIKKOJA	TOIMINTA ON NÄKYVÄN LAADUKASTA JA SITÄ TUETAAN LAITOKSEN TOIMESTA	TOIMINTA JA LAITOKSEN TUKITOIMINNAT ERITYISEN LAADUKKAITA
Asko Karjalainen draft. 2000 Oulun yliopisto. Opetuksen kehittämissyksikkö	NOVIISI 0+	HERÄÄVÄ 1	KEHITTYVÄ 2	PITKÄLLE EDISTYNYT 3
1) tentit ja oppimisen arviointi	Tentit toteutetaan perinteisen rutiinin pohjalta. Arviointi käsitetään jokseenkin pelkästään kontrollin ja valvonnan kysymyksenä	Yksittäisillä opettajilla on tenttikokeiluja, tenttejä koskevasta palautteesta ollaan kiinnostuneita ja tenttijärjestelmän muodollisia puolia (tenttimahdollisuudet jne.) kehitetään opiskelijaystävällisiksi.	Monipuolinen tenttikäytäntö. Tenttien kehittämistä käsitellään laitostasolla pedagogisilla periaatteilla. Laitos ymmärtää miten tärkeä asia tentit ovat oppimisen ohjaamisen kannalta.	Monipuolinen pedagogisesti hyvin perusteltu tenttikäytäntö. Opettajat valmennetaan tenttien suunnitteluun ja toteutukseen. OKTR on laatinut laitokselle tenttikäytäntöä koskevan kehittämissuunnitelman.
2) opetus suunnitelma sekä koulutusohjelmien rakentaminen	Laitos enempää kuin opettajatkaan eivät ole selviillä kokonaisuudesta. Toimivuutta ei seurata.. Opiskelijapolun rakentumisesta ei käsitellä.	Kurssien ydinainesanalyysia ja kuormittavuusmäärittelyä sekä kurssien välisten yhteyksien kehittämistä tehdään vapaaehtoisuudella. Laitos passiivisen myönteinen. Otetaan selvää opiskelijapolun etenemisestä..	Kurssien ydinainesanalyysia ja kuormittavuusmäärittelyä sekä kurssien välisten yhteyksien kehittämistä tehdään kattavasti. Laitos edellyttää (ja varmistaa toimivuuden) opettajilta jatkuvaa kurssien sisältöjen kehittämistä, opintopolun esteiden poistamista ja kokonaisuuskohtaisen toimivuuden varmentamista.	Laitoksella on selkeä opetus suunnitelman kehittämisen periaateohjelma, jota toteutetaan kattavasti. Opetussuunnitelmassa toteutuu life-long- learning periaatteen mukainen joustava tiedeyhteisöllistä lähtökohdista kehittyvä ydinaines rakenne.
3) opetusmuodot (luennot, harjoitukset jne.)	Opetusmenetelmiä ei tietoisesti pohdita. Opetus toteutetaan vanhalta pohjalta, rutiinimaisesti.	Yksittäiset opettajat kokeilevat tilanteisiin sopivia uusia menetelmiä, opettelevat erilaisia menetelmiä, ottavat selvää mahdollisuuksista.	OKTR ottanut opetusmenetelmien kehittämisen tukitoimet huolekseen. opetusmenetelmiä pohditaan yhteisesti laitostasolla. Opettajia rohkaistaan kokeiluun.	Tilanteenmukaisen pedagogisen opetusmenetelmasoveltamisen periaate ymmärretty, jonka seurauksena monipuolinen ja tarkoituksenmukainen menetelmävaranto. Opettajien opetusmenetelmätietoutta tuetaan systemaattisesti materiaalien ja koulutuksen avulla. Laitos seuraa oman alan opetusta käsitteleviä julkaisuja.
4) tutortoiminta ja yhteistyö opiskelijoiden kanssa	Ei lainkaan opettajien toteuttamaa tutorointia. Vain opiskelijoiden toteuttamaa pienryhmäohjausta ja omaehtoista nuorempien ja muiden opiskelijatovereiden ohjausta esiintyy.	Yksittäisten opettajien varassa olevaa tutorointia esiintyy. Jatkuvuutta ei taata.	Laitos sitoutunut turvaamaan tutoroinnin jatkuvuuden. Opettajille järjestetään edellytykset tehdä tutortoimintaa. Tutoreille varataan aika 1600 tunnista.	Tutorointi on otettu osaksi arkista toimintaa, opetusta ja opinto-ohjausta. Tutorointi on yksi opetusmenetelmä ja se yhdistää kaikkia opetusmenetelmiä. Sitä ei koeta erilliseksi toiminnaksi ja sen jatkuvuus on itsestäänselvyys.
5) opinto-ohjaus	Vastaanottoaikojen varassa. Ei toimivia vastuuhenkilöitä.	Vastuuhenkilöihin panostetaan. Ohjaus selkeästi osoitetaan tehtäväksi ja sitä seurataan.	Opinto-ohjaus ymmärretään koko laitoksen henkilökunnan tärkeksi tehtäväksi ja osaksi opiskelijapolun edistämistä. Vastuuhenkilöiden työtä arvostetaan näkyvästi ja tutoroinnin avulla vastuu jaetaan mahdollisimman laajalle.	Opinto-ohjausta on tutorointiin yhdistyneenä riittävästi kaikissa opiskelijapolun vaiheissa. Tutorit ja nimetyt opinto-ohjaajat toimivat hyvässä yhteistyössä. Laitoksella on selkeä opinto-ohjauksen strategia, jota toteutetaan systemaattisesti ja seurataan vuosittain.
6) opetuksen arvostus ja opetusansioiden huomioon ottaminen laitospöytäkirjoissa, henkilöstöstrategiassa (MERITOINTIJÄRJESTELMÄ)	Henkilövalinnoissa painavat vain tutkimusansiot. Kielteinen tai väheksyvä asenneilmapiiri opetustyön meritoituvuuden suhteen.	Opettajilla on portfolioita ja niitä käytetään viranhaussa. On kuitenkin hämärää ja epäselvää, millä tavoin ne arvioidaan tai mikä laitoksen asenne opetusansioihin on.	Laitos on laatinut selkeät periaatteet opetusansioiden tasavertaiselle huomioon ottamiselle sekä portfolioiden arvioinnille ja toteuttaa niitä. Portfolioiden tekemiseen kannustetaan ja asenneilmapiiri opetusansioiden meritoituvuutta kohtaan on myönteinen.	Laitosjohto on selkeästi sitoutunut siihen, että opetusansiot ja hyvä opetustyö otetaan tasavertaisesti huomioon. Käytäntö on luonteva, toimivuudesta on jatkuvaa näyttöä ja laitos seuraa toteutumista. Opettajat perehdytetään hyvin opetustyöhön ja heille annetaan siitä palautetta kehityskeskusteluissa vuosittain. Portfolio on kehityskeskustelun pohjana.

7) tutkimuksen ja opetuksen yhteenliittäminen, toisiaan tukevuus	Koulumainen ja oppikirjamainen opetuskäytäntö. "Yliopisto lukion jatkeena". Nähdään mahdollisena tieteellinen työtapana perusopetuksessa.	Laitoksella on tutkimusryhmiä ja /tai yksittäisiä opettajia, jotka kytkevät opetustyön ja tutkimustyön yhteen.	Laitos ottanut huolekseen varmistaa sen, että heti opintojen alusta lähtien opiskelijat saavat jonkinlaista kontaktia laitoksen tutkimustyöhön. Uudelle tiedolle tehdään sijaa opetusohjelmissa. Opettajia kannustetaan integroimaan opetusta omaan tutkimustyöhön ja laitoksen projekteihin.	Opiskelu ja tutkimustyö integroitu mahdollisimman läheisesti koko opiskelijapolun ajan. Laitoksella on selkeä visio tiedeyhteisöllisestä opetuskäytännöstä. Opiskelija ymmärretään tiedeyhteisön jäseneksi, häntä tuetaan alusta lähtien tietoisesti tästä näkökulmasta.
8) oppimateriaali	Kyhättyä, kasailtua. Monisteita, kalvoja dioja tms. ilman yhtenäistä linjaa ja pedagogista perustelua.	Yksittäisillä opettajilla huomattavaa kehittämistä opetusmonisteiden, oppikirjojen yms. muodossa. Laitos ei kiinnitä asiaan huomiota.	OKTR ottanut laitoksen oppimateriaalin tason ja toteutustavan tukemisen tehtäväkseen. Pedagogiseen puoleen panostetaan ja oppimateriaalin taitto/painatus/jakelu tehokkaasti toteutettu. Opettajia tuetaan ja ohjataan.	Koko laitostyö osallistuu oppimateriaalin kehittämiseen. Opiskelijat integroitu mukaan toimintaan. Materiaalien yhteissuunnittelu ja yhteiskäyttö itsestään selvä periaate.
9) opetuksen jatkuva laadunarviointi ja opiskelijapalautejärjestelmät / muut palautejärjestelmät	Mitään palautetta ei kerätä koko laitoksen mitassa. Yksittäiset opettajat voivat kerätä omiin tarkoituksiinsa. Laitostason asenneilmapiiri varsinkin opiskelija-arvioita kohtaan kielteinen tai väheksyvä.	Opiskelijapalautejärjestelmää yritetään pitää yllä. Jatkuvuus kuitenkin epävarmaa, sillä opiskelijat ovat passiivisia, eivätkä anna palautetta. Palautetta ei ehditä / osata / haluta hyödyntää kehittämisessä. Palautteen keruu tiedetään tärkeäksi mutta koetaan ahdistavaksi. Nähdään tärkeänä valmistuneiden ja työelämäpalaute, jota kuitenkin ei kerätä.	OKTR on ottanut tehtäväkseen varmistaa sen, että palautejärjestelmä aina toimii vaikeuksista huolimatta. Jatkuvaa kehittämistä toteutetaan ja opiskelijat otetaan mukaan toimintaan. Palautea arvostetaan ja se otetaan huomioon toiminnassa, joka myös säännöllisesti raportoidaan opiskelijoille. Palautepäiväkäytäntö pidetään yllä ja sen arvo ymmärretään. Työelämäpalautea kerätään ainakin epämuodollisesti alumneilta ja avainyhteyksien kautta.	Laitosjohto on selkeästi ilmaissut että palauteprosessien hyvä toiminta on laitoksen ja oppiaineiden kunnia-asia. Palauteita täytyy saada ja se täytyy viedä toimenpiteiden tasolle. Laitoksella on turvallinen ja luottamuksellinen ilmapiiri opettajien ja opiskelijoiden välillä, jonka seurauksena kritiikki voi olla kovaa, mutta se osataan aina ottaa rakentavana. Laitoksella on selkeä strategia niin opiskelija- kuin työelämäpalauteenkin keruussa ja hyödyntämisessä. Myös sisäistä kollega-auditointia käytetään.
10. yhteistyö muiden laitosten kanssa opetuksen järjestämisessä (myös muiden yliopistojen kanssa)	Rajoittuu pakolliseen minimiin erilaisissa hallintoelmissä ja sivuaineasiakkuuksissa. Yksittäisillä opettajilla ja tutkijoilla voi olla hedelmällistä yhteistyötä yli laitosrajojen.	Neuvotteluja pyritään käymään ja käynnistämään sivuaineopetuksen ja muun yhteistyön, tilojen laitteiden tms. puitteissa laitosjohdon ja avainhenkilöiden kesken. Helposti hiipuvaa ja satunnaista.	Laitos (OKTR) on laatinut selkeän yhteistyön ja sisäisen asiakkuuden strategian, jota laitos toteuttaa. Yhteistyötä haetaan ja pyritään tarjoutumaan avuksi niin opetuksen kuin tutkimuksenkin alueella.	Verkostoituminen lähitieteiden ja naapurilaitosten kanssa toimiva ja itsestään selvä työtapana. Palvelujen ja osaamisen vaihtoa ilman sisäistä laskutusta. Merkittävää yhteistyötä koulutuksen alueella myös muiden kotimaisten yliopistojen kanssa.
11. opetusteknologian käyttö ja kehittäminen	Laitos ei panosta mitenkään, riippuu yksittäisistä opettajista.	Opettajilla on selkeää ja näkyvää innostusta uusien medioiden ja teknologiatuettujen oppimisympäristöjen käyttöön. Laitos passiivisen positiivinen.	Laitoksella suuri innostus asiaan, ehkä yliampuvaakin toimintaa, laitos panostaa sekä materiaaliin että henkisiin resursseihin.	Opetusteknologian käyttö ja kehittäminen ymmärretään yhdeksi tärkeäksi työkaluksi ja laitoksella on selkeä, käytännössä toimiva (OKTR:n laatima) visio ja strategia tietotekniikan pedagogisesta käytöstä ja merkityksestä.
12. opiskelijoiden keskeyttäminen, valmistuminen ja työllistyminen	Ei seurata laitostasolla.	Laitos seuraa satunnaisesti lähinnä reagoitujen (mutta selkeästi reagoitujen) tiedekunnan ja keskushallinnon tuottamiin tilastoihin.	Laitos seuraa itse opiskelijoiden etenemistä opiskelijapolulla lukukausittain. Ryhtyy välittömästi ja käskemättä toimenpiteisiin.	Jatkuva monitorointi näkyy käytännössä. Opiskeluajat tavoitteen mukaiset, valmistumisprosentti korkea. Laitosjohto on sitoutunut hyvän tuloksen ylläpitämiseen.
13. työelämäyhteydet ja yhteistyö ulospäin koulutuksen kehittämisessä ja järjestämisessä	Ei laitostason yhteyksiä. Asenne ennakkoluuloinen tai kielteinen.	Opettajilla runsaasti yhteyksiä työnantajiin ja yritysalamään. Ei hyödynnetä laitostasolla palautekanavana eikä opetussuunnitelmallisesti.	Alumniverkostoa ylläpidetään. Opiskelijoiden harjoittelua hyödynnetään palautekanavana ja järjestetään mentoryhteyksiä.	Laitoksella on osana opetussuunnitelmaa ja opetussuunnitelmataytä systemaattinen verkostoituminen työelämään ja intressipiireihin. Opiskelijoiden työssäoppiminen ja työn ohella opiskelu kitkatonta ja joustavaa.
14. opettajien korkeakoulupedago- ginen kouluttautuminen	Opettajia ei päästetä koulutukseen. Jotkut kouluttautuvat salaa kollegoilta ja laitosjohdolta omilla kustannuksillaan.	Opettajat saavat vapaasti osallistua koulutukseen. Laitos passiivisen myönteinen ja osallistuu koulutuksesta aiheutuviin kustannuksiin.	Laitos tukee opettajien kouluttautumista. Kehottaa ja kannustaa osallistumaan koulutukseen. Hoitaa kustannukset täysimääräisesti.	Laitosjohto edellyttää kouluttautumista ja on laatinut opettajille henkilökohtaisen kouluttaumissuunnitelman, jonka toteutumista seurataan kehityskusteluissa vuosittain.
15. laitoksen opetuksenkehittämissä ryhmän toiminta, ohjelma, suunnitelma, strategia opetuksen kehittämiseksi	Ryhmä hajalla tai hiipunut. Kokoontuu satunnaisesti esimerkiksi opiskelijoiden painostuksesta.	Ryhmä kokoontuu joksikin säännöllisesti. Ei selkeää toimintastrategiaa eikä kuvaa tehtävistä tai vastuusta.	Ryhmä ymmärtää vastuunsa ja tehtävänsä ja on laatinut itselleen selkeät toimintaperiaatteet. Ryhmällä on selkeä rooli laitoksen oppimisen ja opetuksen tukitoimien suunnittelussa ja kehitystyössä.	Ryhmä johtaa laitoksen opetuksen ja opetusohjelmien kehittämistä hyvässä yhteistyössä laitosjohdon kanssa. Laatii laitokselle opetuksen kehittämissä strategian (yhdessä OKY:n kanssa) ja seuraa sen toteutumista ja toimeenpanoa.
16. opettajien yhteistyö	Satunnaista ja henkilökohtaihin pohjautuvaa. Ei suosita eikä	Spontaaneja tiimejä ja työpareja huomattavissa määrin.	Laitos kannustaa yhteistyöhön. Asiaa käsitellään henkilökunnan kesken ja yhteisissä	Laitos on laatinut opetussuunnitelmaan pohjautuvan selkeän

opetuksen järjestämisessä ja toteutuksessa	suositella.		tilaisuuksissa.	yhteistyöohjelman, jossa toisiinsa liittyvien kurssien opettajat tekevät tarvittavassa määrin yhteissuunnittelua, auditointia ja yhteistä opetustyötä.
17. kansainvälistymisen hyödyntäminen opetusjärjestelyissä	Hallinnon vaatima minimipanostus. Opiskelijavaihtoa enemmän ulos- kuin sisäänpäin.	Opiskelijavaihto toimii molempiin suuntiin. Koetaan jossakin määrin rasitteena. Vieraskielisen opetustarjonnan kattavuudessa ja jatkuvuudessa vaikeuksia.	OKTR on laatinut laitokselle selkeän KV-strategian. Kansainvälisestä toiminnasta pyritään saamaan sisällöllistä hyötyä sekä opetukselle että tutkimukselle.	Laitos on mielekkäällä tavalla verkostoitunut yhteistyön kannalta tärkeiden ulkomaisten yliopistojen kanssa. Opiskelijavaihto ja opettajien liikkuminen on luontevaa ja arkipäiväistä. Laitoksella on yhteistä opetusta ja mahdollisesti yhteisiä kursseja, opintokokonaisuuksia tai jopa koulutusohjelmia ulkomaisten yhteistyöyliopistojen kanssa.
18. Laitoskohtainen kouluttautuminen	Tuntematon käsite.	Laitoksen opettajat hyödyntävät OKY:n tai henkilösopeluiden laitospalveluita satunnaisesti työhönsä, pikakoulutusten ja konsultoinnin alueilla.	Laitosjohto/OKTR tilaa itselleen räätälöityä koulutusta OKY:ltä, henkilösopeluilta tai yliopiston ulkopuolelta.	Laitoksella on selkeä ja yhteisesti laadittu sisäinen kouluttautumisen ja henkilöstön kehittämisen strategia ja toimintaohjelma, jota toteutetaan. Myös opiskelijat ovat mukana laitosta eheyttävässä ja ilmapiiiriä kehittävässä toiminnassa.
19. Miten tämä raportti syntyi	Amanuessi tai muu toimeksiannon saanut kirjoitti sen. Suunnattu hallinnolle.	Amanuessi tai muu toimeksiannon saanut kirjoitti sen toteuttaen aineistonkeruun yhdessä laitosjohtajan kanssa. Suunnattu pääosin hallinnolle. Epämääräinen käyttö, henkilökunta ja opiskelijat eivät selkeästi saa tiedokseen.	OKTR koordinoi raportin laadinnan. Se kirjoitettiin yhteisesti (tai yhteistyötä synnyttävästi) opettajien ja opiskelijoiden kanssa ja se on suunnattu ensisijaisesti laitoksen opetuksen kehittämistä palvelemaan. Opettajat/opiskelijat saavat raportin tiedoksi heti sen valmistuttua. Raportti julkaistaan WWW:ssä vähintäinkin yliopiston sisäisesti luettavaksi.	Raportti syntyi normaalin vuosittaisen seurannan sivutuotteena OKTR:n toiminnassa yhdessä laitosjohtajan ja opiskelijoiden kanssa laitoksen opetuksen kehittämiseksi ja sisäistä tietoisuutta luomaan. Opettajat/opiskelijat ja henkilökunta ovat koko ajan tietoisia asiasta. Raportti julkaistaan WWW:ssä vähintäinkin yliopiston sisäisesti luettavaksi.