

Opiskelijoiden hyödyntäminen tietojenkäsittelytieteen opettamisessa

Hanna Arpiainen

Aine
Helsingin Yliopisto
Tietojenkäsittelytieteen laitos

Helsinki, 9. maaliskuuta 2013

Tiedekunta — Fakultet — Faculty		Laitos — Institution — Department	
Matemaattis-luonnontieteellinen		Tietojenkäsittelytieteen laitos	
Tekijä — Författare — Author Hanna Arpiainen			
Työn nimi — Arbetets titel — Title Opiskelijoiden hyödyntäminen tietojenkäsittelytieteen opettamisessa			
Oppiaine — Läroämne — Subject Tietojenkäsittelytiede			
Työn laji — Arbetets art — Level Aine	Aika — Datum — Month and year 9. maaliskuuta 2013	Sivumäärä — Sidoantal — Number of pages 9	
Tiivistelmä — Referat — Abstract Aine.			
Avainsanat — Nyckelord — Keywords avainsana 1, avainsana 2, avainsana 3			
Säilytyspaikka — Förvaringsställe — Where deposited			
Muita tietoja — Övriga uppgifter — Additional information			

Sisältö

1	Johdanto	3
2	Oppilaat opettajina	3
2.1	Ohjaajien työtehtäviä	4
2.2	Millainen on hyvä ohjaaja?	4
2.3	Ohjaajien koulutus	5
2.4	Käytännön organisaatio	5
3	Seurauksia	6
4	Yhteenveto	7
5	Lähteet	8

1 Johdanto

Tietojenkäsittelytieteen opiskelijoiden lukumäärän vaihtelu on syklistä [10]. Esimerkiksi kotitietokoneiden ja World Wide Webin yleistymisen kasvattivat alan suosiota ja opiskelijoiden määrä kasvoi, kun taas IT-kuplan puhkeaminen 2000-luvun alussa vähensi opiskelijamäärää. Ajoittaisista taantumisista huolimatta opiskelijoiden määrä on kuitenkin pitkällä ajanjaksolla kasvanut, ja teollisuus tarvitsee yhä enemmän osaaajia. Muiden aineiden opiskelijat haluavat myös opiskella tietojenkäsittelytieteen perusteita, koska he uskovat, että ohjelmointitaidot auttavat heitäkin työmarkkinoilla. Etenkin alkeiskursseille pitäisi siis pystyä järjestämään riittävästi opetusta.

Usein ongelmana on taloudellisten resurssien puute, joten opiskelupaikkojen lukumäärää voidaan joutua rajoittamaan. Lisäksi etenkin 1980-luvulla osaavista opettajista oli pulaa, joten henkilökuntaan jouduttiin palkkaamaan riittämättömästi koulutettuja ulkopuolisia [10]. Suurten ryhmäkokojen seurauksena opiskelijat uhkaavat hukkaa massaan ja voivat sen seurauksena menettää mielenkiintonsa [3].

Ongelmaa voi ratkaista kehittämällä yhteistyötä teollisuuden kanssa, koska sillä on taloudellisia resursseja tukea opetusta [10]. Toinen ratkaisu on palkata alkeiskursseille mahdollisimman pätevää henkilökuntaa.

Kolmas ratkaisu ongelmaan on käyttää opiskelijoita opetuksessa, mihin tutkielma keskittyy. Monet yliopistot ovat päättäneet hyödyntää opiskelijoita kurssien opetukseen, sillä he ovat halvempia kuin pitkälle koulutettu työvoima. Taloudellisen edun lisäksi opiskelijoiden on myös havaittu olevan hyödyllisiä monella muullakin tavalla.

Vaikka opiskelijoita on hyödynnetty tietojenkäsittelytieteen opetuksessa jo pitkään, aihetta on tutkittu niukasti.

2 Oppilaat opettajina

Opiskelijat voivat toimia ohjaajana ryhmälle alkeiskurssin opiskelijoita [7]. Ohjaajat tarjoavat lisäopetusta kurssin varsinaisten luentojen tueksi ja auttavat kurssin opiskelijoita tehtävien kanssa.

Ryhmänohjaajamallin lisäksi opiskelijoita voi käyttää opetuksessa myös vertaismentoreina [12]. Vertaismentorimallissa ne opiskelijat, jotka tarvitsevat lisää tukea opintoihinsa tai tarvitsevat ylimääräisiä haasteita saavat mentoriohjaajan, joka osaa kurssin sisällön hyvin. Mentori tapaa neuvottavaansa ja avustaa tehtävien kanssa. Mentorointi voi mennä myös ristiin; opiskelija voi olla mentorina toiselle yhdellä kurssilla ja neuvottavansa mentoroitavana jollakin toisella.

2.1 Ohjaajien työtehtäviä

Ryhmänohjaajien tehtäviin kuuluu esimerkiksi viikottaisten keskustelusesioden pitäminen [7]. Keskustelusesiossa käydään läpi kurssin sisältöä, mutta sen ei tarvitse noudattaa mitään tiukkaa kaavaa, vaan ryhmänohjaaja voi muokata sen vastaamaan ryhmänsä tarpeita. Ryhmänohjaaja voi esimerkiksi kerrata luentojen asioita, esittää lisäesimerkkejä tai vastata opiskelijoiden esittämiin kysymyksiin. Viikkotapaamisten ei tarvitse olla vain ohjaajan esiintymistä, vaan ne voivat sisältää opiskelijoiden ryhmäkeskusteluja [1]. Tarvittaessa ryhmänohjaajan on oltava valmis hylkäämään aiemmat suunnitelmansa, mikäli selviää, että ryhmä on ymmärtänyt jonkin tärkeän asian väärin ja tilanne on korjattava nopeasti [7].

Ryhmänohjaajan tehtäviin voi kuulua viikottainen yksilöllinen palautekeskustelu jokaisen ryhmänsä opiskelijan kanssa. Opiskelija voi selittää ratkaisunsa johonkin tehtävään, tai kertoa, että ei ole ymmärtänyt jotakin kurssin asiaa. Ohjaaja taas voi selittää, mistä jokin virhe johtuu, ja miten sen voi jatkossa välttää. Näissä keskusteluissa ryhmänohjaaja saa kuvan opiskelijan edistymisestä kurssilla [7, 6].

Ryhmänohjaajat voivat päivystää tietokoneluokassa valmiina auttamaan kurssin opiskelijoita kurssiin liittyvissä ongelmissa. Ruuhka-ajoille päivystäviä ohjaajia pyritään järjestämään enemmän [7, 6].

Ohjaajat voivat pitää kurssin kertaustilaisuuden ennen koetta. Kertaustilaisuudessa he voivat kertoa, mitkä kurssin asiat ovat tärkeitä ja mitä kokeessa todennäköisesti kysytään, tai esitellä vanhoja kokeita ja niiden ratkaisuja [1]. Usein ohjaajan tehtäviin kuuluu tehtävien tarkistaminen. Ohjaaja voi myös päivittää kurssin verkkosivuja [2]. Ohjaajan voidaan edellyttää osallistuva kurssin luennoille pysyäkseen mahdollisimman hyvin perillä kurssin etenemisestä [6].

2.2 Millainen on hyvä ohjaaja?

On vaikea määritellä tarkkaan, millainen on hyvä ohjaaja, koska niin monet erilaiset ohjaajat ovat onnistuneet tehtävässään. Ohjaajalta vaaditaan kuitenkin aitoa innostusta auttaa muita oppimaan ja vastuullisuutta noudattaa kurssin käytäntöjä. Ohjaajan ei ole välttämätöntä olla ohjelmoinnin erityisosaaja, mutta hänen edellytetään ymmärtävän kurssin sisältö ja olevan valmis kehittämään itseään pystyäkseen auttamaan opetettaviaan riittävästi [7]. Ohjaaja voi jopa hyötyä siitä, että hänellä on itsellään ollut vaikeuksia oppia ohjelmoimaan, sillä se voi auttaa häntä ymmärtämään paremmin opettaviensa vaikeuksia samojen tehtävien kanssa [1].

On havaittu, että uudet opiskelijat ovat yleensä parempia ohjaajia kuin vanhemmat, opinnoissaan pidemmälle edenneet opiskelijat [2]. Uusilla ohjaajilla on muistissaan hyvä kuva kurssin pääasioista, koska he ovat itse käyneet kurssin vasta vähän aikaa sitten. Pitkälle edenneiden opiskelijoiden taas

saattaa olla vaikeaa ymmärtää kurssin materiaalia, opetustekniikkaa tai alkeiskurssien opiskelijoiden ongelmia, ja heidän voi olla vaikeampi ymmärtää laitoksen tietokoneita [7]. Ohjattavien voi olla vaikea suhtautua ohjaajaan, jos tämä on heitä kovin paljon vanhempi [1].

Aito kiinnostus opetettavien auttamiseen on keskeistä ohjaajan menestykselle. Ohjaaja, joka esimerkiksi on töissä vain palkan vuoksi ja pyrkii pelkkään minimisuoritukseen, ei valmistaudu opetustilaisuuteen riittävästi tai vähättelee opetettaviaan tai opetettavaa materiaalia ei tue oppimista [8].

2.3 Ohjaajien koulutus

Uusien ohjaajien kouluttamiseen voi kuulua erillinen tietojenkäsittelytieteen opettamiseen keskittyvä kurssi [7, 9]. Myös teollisuuden edustajat voivat olla kiinnostuneita kouluttamaan ohjaajia [4].

Ohjaajien kouluttamiseen liittyy usein keskustelutilaisuuksia. Niissä voidaan esimerkiksi harjoitella tehtävien pisteyttämistä, hankalien käsitteiden opettamista tai hankalien opiskelijoiden käsittelyä [6]. Kokeneet ohjaajat voivat osallistua uusien ohjaajien koulutukseen kerratakseen omia taitojaan, ja samalla he voivat jakaa kokemustaan ja toimia roolimalleina uusille ohjaajille [7].

Kehittyäkseen työssään ohjaajien on tärkeää saada palautetta [5]. Myönteinen palaute parantaa ohjaajan itseluottamusta, mikä puolestaan parantaa hänen opetustaitojaan.

2.4 Käytännön organisaatio

Ohjaajia hyödyntävien kurssien henkilökunnan rakenne vaihtelee kurseittain ja yliopistoittain. Esimerkiksi pienessä yliopistossa voi toimia malli, jossa luennoijan lisäksi kurssin henkilökuntaan kuuluu vain muutama ohjaaja [2], kun taas suuremmissa laitoksissa kursseilla, joilla on paljon opiskelijoita, on hyödyllistä, että kurssin henkilökuntaan kuuluu ohjaajien työstä vastaava assistentti [6]. Assistentti toimii ohjaajien ja luennoijan välisenä kontaktina ja jakaa kurssin materiaalin eteenpäin ohjaajille. Hän voi auttaa luennoijaa kurssin sisällön kanssa. Assistentti voi olla joku kurssin ohjaajista tai erillinen tehtävään palkattu henkilö.

Mikäli kurssilla on paljon opiskelijoita ja henkilökuntaa, ohjaajamalli tarvitsee usein toimiakseen jonkinlaisen koordinaattorin. Koordinaattori voi olla esimerkiksi opinnoissaan pitkälle edennyt opiskelija, joka on aiemmin toiminut ryhmänohjaajana. Suuremmilla kursseilla voidaan tarvita useampia koordinaattoreita [9].

Koordinaattorin tehtäviin kuuluu kurssin käytännön hallinnointi, eli esimerkiksi luokkahuoneiden varaaminen ja tarvittavien ohjaajien jakaminen kursseille. He myös hoitavat ohjaajien valinnan, palkkaamisen ja kouluttamisen [7, 9]. Koordinaattori voi huomauttaa ohjaajan puutteellisesta toiminnasta

[7].

Koordinaattorien kuuluu ylläpitää tiedon kulkua ja kommunikointia kurssilla. Heidän tulee myös järjestää tarvittavat kommunikointikanavat opiskelijoiden ja ohjaajien välille [7]. Koordinaattorien tehtäviin kuuluu järjestää tapaamisia kurssin henkilökunnalle, jotta luennoija saa kuvan opiskelijoiden etenemisestä. Viikkopalaverissa ohjaajat voivat kertoa mahdollisista ongelmista, ja luennoija voi tarpeen tullen hidastaa tahtia tai selittää jonkin epäselväksi jääneen asian uudestaan paremmin. Palaverissa ohjaajat pääsevät tapamaan toisiaan ja jakamaan tietoa. Koordinaattorien kuuluu myös järjestää tapaamisia, joissa käydään läpi opetukseen liittyviä asioita [7, 9].

3 Seurauksia

Opiskelijoiden käyttämisestä ohjaajina on taloudellista etua yliopistolle, sillä heille maksettava palkka on yleensä matalampi kuin kokeneemmille assistenteille, tai heidät palkitaan opintopisteillä. Ohjaajien palkitseminen opintopisteillä palkan sijaan on kuitenkin jakanut mielipiteitä. Vaikka muillakin tieteenaloilla opiskelijoita palkitaan opintopisteillä tutkimusprojekteihin osallistumisesta, on esitetetty, että opettamiseen osallistuminen ei olisi yhtä arvokasta kuin tutkimukseen osallistuminen [7].

Lakisääteiset syyt voivat rajoittaa opiskelijoiden hyödyntämistä opetuksessa. Säädökset voivat esimerkiksi kieltää opiskelijoita arvostelemasta saman asteen opiskelijoiden tehtäviä varmistaakseen, että yliopistot eivät laiminlyö velvollisuuksiaan arvostelussa. Säädökset voivat myös kieltää liian pienet kurssit. Vaikka jollakin tietojenkäsittelytieteen alkeiskurssilla olisi valtavat määrät opiskelijoita, jos heidät on jaettu pieniin ryhmiin, voi yksi iso kurssi tilastollisesti näyttää monelta pieneltä [6].

Yliopistolla työskentelevät opiskelijat voivat toimia roolimalleina muille opiskelijoille [9, 12], sillä opiskelijoiden voi olla vaikea nähdä itsensä esimerkiksi kokeneena luennoijana, kun taas ohjaaja tarjoaa helpommin lähestyttävämmän roolimallin [11]. Jos ohjaajat valitaan opiskelijoiden keskuudesta, voidaan henkilökunta saada paremmin vastaamaan opiskelijoita. Etenkin naisille ja vähemmistöjen edustajille voi olla tärkeää saada kaltaisansa roolimalli, jotta heidän on helpompi tuntea kuuluvansa joukkoon [4]. Ryhmänohjaajamalli mahdollistaa paremmin pienryhmissä opiskelun, joka auttaa opiskelijoita tutustumaan toisiinsa. Yhteenkuuluvuuden tunne saa opiskelijat pysymään alalla paremmin.

Ryhmänohjaajamalli voi toimia koulutuksena ja innoituksena uusille luennoitsijoille yliopiston tulevaisuudessa [9, 4]. Ryhmänohjaajia ohjaavat opettajat saavat tilaisuuden vaikuttaa ohjaajien näkemykseen opetusalaista [4].

Koska opiskelijoiden palkkaaminen ohjaajiksi on halvempaa kuin muiden vaihtoehtojen, voi heitä palkata enemmän. Näin opetettavat saavat enemmän yksilöllistä huomiota. Kurssilla voi olla enemmän pieniä tehtäviä, jotka innos-

tavat opiskelijoita paremmin lukemaan materiaalia, kun tehtävien tarkastamiseen on käytössä enemmän henkilökuntaa [2]. Jonkun tietyn henkilökunnan edustajan mahdollinen puolueellisuus ei välttämättä pääse vaikuttamaan opiskelijoihin, jos luokassa on samaan aikaan useita erilaisia ohjaajia [4]. Kurssin ilmapiiriä rentouttaa, jos opiskelijat näkevät luennoijan ja ohjaajan tulevan hyvin toimeen keskenään. Mikäli ohjaaja on läsnä esimerkiksi luennolla ja esittää luennotsijalle kysymyksiä aiheesta, rohkaisee se opiskelijoitakin osallistumaan enemmän luennoilla. Kurssin opiskelijat hyötyvät, jos heidän keskuudessaan on ohjaaja, joka osaa heti kurssin alussa kertoa luennoijan opetustyylistä ja painotuksista. Ohjaaja voi olla jonkin tietojenkäsittelyn osa-alueen erityisosaaja ja tuoda osaamisensa kurssin hyödyksi. Motivoitunut ohjaaja motivoi opetettaviakin työskentelemään ahkerammin [2]. Kurssin opiskelijoiden voi olla helpompi antaa palautetta ohjaajalle kuin luennoijalle [4]. Ohjaajat pystyvät antamaan hyödyllistä palautetta kurssin toiminnasta ja antaa parannusehdotuksia [1].

Vaikka ohjaajat pääasiassa hyötyvät siitä, että he ovat suurin piirtein saman ikäisiä kuin opetettavansa, voi siitä seurata myös ongelmia tehtävien arvostelun suhteen. Koska ohjaajalla ei mahdollisesti ole paljon kokemusta tai auktoriteettiä, voi hänen olla vaikea käsitellä opiskelijoita, jotka ovat mielestään saaneet liian huonon arvosanan. Ohjaajilla pitäisikin olla selvät ohjeet tehtävien pisteyttämisen suhteen kiistatilanteiden varalta [9]. Toisaalta jos pidetään tärkeänä sitä, että opiskelijat mieltävät ohjaajat vertaisekseen, voi olla parempi, että ohjaajat eivät osallistu tehtävien pisteyttämiseen ja opiskelijoiden arvosteluun [4].

Ohjaajat saavat palkkaa tai opintopisteitä, minkä lisäksi opettaminen parantaa heidän ryhmätyö- ja esiintymistaitojaan. Opettaessaan he oppivat itsekin materiaalin paremmin [6], ja opetettavien kannustaminen hyvään ohjelmointityyliin parantaa ohjaajankin ohjelmointityyliä [9]. Ohjaajana toimiminen opettaa vastuukantoa ja liittää opiskelijan paremmin mukaan yliopiston yhteisöön [2]. Ohjaajat voivat saada toisistaan seuraa ja osallistua keskenään erilaisiin vapaa-ajan aktiviteetteihin [9].

4 Yhteenveto

Useissa yliopistoissa on todettu, että opiskelijoiden käyttö opetuksessa nostaa opetuksen laatua ja mahdollistaa suurempien opiskelijamäärien opettamisen ilman, että kustannukset nousevat. Ohjaajana toimiminen antaa opiskelijalle hyödyllisiä taitoja ja kokemusta, ja ohjaajien läsnäolo kursseilla rentouttaa kurssin ilmapiiriä. Mallin on havaittu toimivan niin suurissa kuin pienissäkin oppilaitoksissa.

5 Lähteet

- [1] Decker, Adrienne, Phil Ventura ja Christopher Egert: *Through the looking glass: reflections on using undergraduate teaching assistants in CS1*. SIGCSE Bull., 38(1):46–50, maaliskuu 2006, ISSN 0097-8418. <http://doi.acm.org/10.1145/1124706.1121358>.
- [2] Dickson, Paul E.: *Using undergraduate teaching assistants in a small college environment*. Teoksessa *Proceedings of the 42nd ACM technical symposium on Computer science education*, SIGCSE '11, sivut 75–80, New York, NY, USA, 2011. ACM, ISBN 978-1-4503-0500-6. <http://doi.acm.org/10.1145/1953163.1953187>.
- [3] Kay, David G.: *Large introductory computer science classes: strategies for effective course management*. SIGCSE Bull., 30(1):131–134, maaliskuu 1998, ISSN 0097-8418. <http://doi.acm.org/10.1145/274790.273177>.
- [4] Morgan, J., A. Kenimer, T. Kohutek, J. Rinehart ja M. Lee: *Peer teacher from an instructor's perspective*. Teoksessa *Frontiers in Education, 2002. FIE 2002. 32nd Annual*, nide 3, sivut S2C–11 – S2C–15 vol.3, nov. 2002.
- [5] Patitsas, Elizabeth ja Patrice Belleville: *What can we learn from quantitative teaching assistant evaluations?* Teoksessa *Proceedings of the Seventeenth Western Canadian Conference on Computing Education*, WCCCE '12, sivut 36–40, New York, NY, USA, 2012. ACM, ISBN 978-1-4503-1407-7. <http://doi.acm.org/10.1145/2247569.2247582>.
- [6] Reges, Stuart: *Using undergraduates as teaching assistants at a state university*. SIGCSE Bull., 35(1):103–107, tammikuu 2003, ISSN 0097-8418. <http://doi.acm.org/10.1145/792548.611943>.
- [7] Reges, Stuart, John McGrory ja Jeff Smith: *The effective use of undergraduates to staff large introductory CS courses*. SIGCSE Bull., 20(1):22–25, helmikuu 1988, ISSN 0097-8418. <http://doi.acm.org/10.1145/52965.52971>.
- [8] Richards, L.G.: *Teaching graduate teaching assistants (GTAs) how to teach*. Teoksessa *Frontiers in Education Conference, 2000. FIE 2000. 30th Annual*, nide 2, sivut F3F/14 –F3F/20 vol.2, 2000.
- [9] Roberts, Eric, John Lilly ja Bryan Rollins: *Using undergraduates as teaching assistants in introductory programming courses: an update on the Stanford experience*. SIGCSE Bull., 27(1):48–52, maaliskuu 1995, ISSN 0097-8418. <http://doi.acm.org/10.1145/199691.199716>.

- [10] Roberts, Eric S.: *Meeting the challenges of rising enrollments*. ACM Inroads, 2(3):4–6, elokuu 2011, ISSN 2153-2184. <http://doi.acm.org/10.1145/2003616.2003617>.
- [11] Roberts, Eric S., Marina Kassianidou ja Lilly Irani: *Encouraging women in computer science*. SIGCSE Bull., 34(2):84–88, kesäkuu 2002, ISSN 0097-8418. <http://doi.acm.org/10.1145/543812.543837>.
- [12] Tashakkori, Rahman, James T. Wilkes ja Edward G. Pekarek: *A systemic mentoring model in computer science*. Teoksessa *Proceedings of the 43rd annual Southeast regional conference - Volume 1*, ACM-SE 43, sivut 371–375, New York, NY, USA, 2005. ACM, ISBN 1-59593-059-0. <http://doi.acm.org/10.1145/1167350.1167453>.