Education evaluation in the Department of Computer Science at the University of Joensuu

Jyrki Katajainen

23.1.–31.3.2002; last modification 5.4.2002

How was this evaluation done?

The head of the department contacted me by e-mail 7.11.2001 and asked if I was willing to do this evaluation. I agreed to do the job the following day. I received the official appointment letter from the dean in the beginning of December 2001. The other panel member appointed was Matti Mäkelä, who is an emeritus professor from the University of Helsinki. As required by the faculty, he provided a separate evaluation report.

This evaluation is part of the project driven by the faculty where the goal is to improve teaching and studies in general. The project is carried out according to the guidelines[1] provided by the evaluation office at the University of Helsinki — only in a lighter form.

As a starting point of my work I received a self-evaluation report from the department on 23.1.2002. Next day, after reading the report and some of the other material enclosed, I contacted five randomly selected staff members and asked for a functional description of their work in order to get a better picture of the department. A day later the number of teachers was increased by two to cover all study lines. On 15.2.2002 I used one day to read all the e-mail correspondence so far.

Matti Mäkelä and I visited the department during the period 20.–23.2.2002. In brief, the program was as follows:

Wednesday evening, 20.2.:

- An informal discussion with the head of the department (Jussi Parkkinen).

Thursday morning, 21.2.:

- The head of the department showed us around the department.
- A formal discussion with the head and the deputy head of the department (Jussi Parkkinen and Simo Juvaste).
- An interview of the representative of the software-engineering group (Raimo Rask).

Thursday afternoon, 21.2.:

- A discussion with the self-evaluation work group and the teaching-improvement work group.
- A discussion with eight voluntary students, most of whom are active in the computer-science student association.

Thursday evening, 21.2.:

- A sauna session with the stem members of the department (Jussi Parkkinen, Marja Kuittinen, Pasi Fränti, Pirkko Voutilainen, and Simo Juvaste).

[1] Helsingin yliopiston koulutuksen ja tutkintojen arviointi 2001–2002, Itsearvioinnin käsikirja, Arviointitoimisto, Helsingin yliopisto (2001).

Friday morning, 22.2.:

- Individual interviews of the randomly selected staff members (Eeva Saukkonen, Ismo Kärkkäinen, Pasi Fränti, and Simo Juvaste).

Friday afternoon, 22.2.:

- Lunch with the dean (Seppo Pasanen) and some of the stem members of the department (Jussi Parkkinen, Marja Kuittinen, and Simo Juvaste).
- A reception where we gave our general impression about the educational situation at the department.
- Two more interviews of staff members (Erkki Sutinen and Markku Hauta-Kasari).

In the meeting room reserved for us some of the recent M. Sc. theses and Ph. D. theses were made available. There were also a small exhibition of lecture notes and textbooks used in the master's program. However, I had little time to acquaint myself with this material.

When doing this evaluation I had two roles to choose: a judge or a consultant. As a docent of the department I found the role of a judge unnatural. Therefore, I saw my main role as a consultant.

I wrote this report on 10. and 28.3.2002, typed it in on 29.3.2002, and finalized it on 31.3.2002. In all, I have used about nine working days for this evaluation. I found it astonishing how well I could learn to know the department in so short time. If there are any errors, they are solely mine of course.

General impression

I had a prejudiced opinion about the department based on my visits in the beginning of the 1990s. At that time the department was led by my former supervisor Martti Penttonen and he was the only person in the whole department that had a Ph. D. degree. Among young scientists the department was considered as a stepping stone for "better" positions in the southern part of Finland. I have to admit that I even myself misused the department in this respect. Therefore, the evaluation visit was eye-opening for me.

The amount of staff members has increased from 17 to 63 in the last five years, the intake of students has four doubled in the same time period, the department has just been moved to a new building, which is connected with a science park not far from the main buildings of the university, and more than 50% of the teaching personal (18) have earned a Ph. D. degree (10). The people I met are energetic and diligent workers. Also, the atmosphere in the department was pleasant. The students we met were all satisfied with their study environment, except some minor details, so my conclusion was that the teachers are doing a good job. The number of Ph. D. students is larger than in our department in Copenhagen even though I expected our department to be larger.

In a book about plans and ideas for a computing department[2], Peter Naur promotes the idea that one should see a university department as a data processing system. I often wonder how seldom we use the tools and methods we teach in our courses in

[2] Peter Naur, *Planer og ideer for datalogisk institut ved Københavns universitet*, Studentlitteratur (1970).

the planning of our own work environment. I was gladly surprised when I heard that at least two of the staff members use the methods in practice.

Recommendation 1 *Use the system analysis and design methods taught to improve the business processes at the department.*

As a concrete example, I can mention the registration of examination results. In the current work process the registration is done twice, once to create a list to the notice board and once to update the student database. In the University of Helsinki the information collected from the students (via a web interface) in connection with their examination registration flows through the system so that the teacher only fills in the grades in the form generated automatically. Such a system has been proposed by instructors — let them implement such a system. There are security problems, but these may stimulate a research area which is not strong in the department at the moment.

To summarize, I got a favourable impression of the department, and the campus is beautifully placed. The staff members and students we discussed with were very co-operative, and I want to thank them all for this.

Recommended reading

When doing this evaluation, I was inspired by the following two evaluation reports done in Sweden and in Denmark, respectively:

Utvärdering av datavetenskapliga/datalogiska utbildningar i Sverige, Högskoleverkets rapportserie 2001:13 R, Högskoleverket (2001). Available at http://www.hsv.se/rapporter_nyhetsbrev/pdf/0113R.pdf.

Evaluering af de videregående datalogiuddannelser, Evalueringsrapport, Evalueringscenteret (1997).

After reading these reports and visiting Joensuu, I can clearly see that the problems in computing departments are about the same in all Nordic countries offering free education. The Swedish report is especially interesting since it lists the practices used for solving the well-known problems. Hence, it can be recommended for the members of the teaching-improvement work group, but also for other decision makers.

I have picked many of the topics directly from the Swedish report to this report. Appendix 1 contains the department information in numbers collected by the self-evaluation work group. In the Swedish report similar numbers are given for each computing department in Sweden so a direct comparison is possible. I recommend that the teaching-improvement work group does this comparison to get a better idea about the quality of teaching processes.

Self-evaluation report

The report was done by a group of 11 people led by Marja Kuittinen. About half of the members were students. I received also a separate evaluation report made by the students and I could see that many of the passages in the final report were directly from that document. The student report was a summary of a web survey carried out by the student association. The results of this web survey were also provided for me even though I did not have time to study the comments carefully. The whole process shows that the students have influence on the department, which is positive.

I expected a more critical self-evaluation report. The report contained little evaluation and it failed to describe the overall structure of the educational system, at least for an external reader. I was most worried about the following five passages.

1. Opetussuunnitelman tekeminen on huomattavasti vaikeutunut viime vuosina opiskelijamäärän lisääntymisen myötä.

The difficulty in putting the teaching programs together was mentioned but not analysed at all. The increase in the amount of the people involved requires more communication so I expected some strategic plans how the situation can be improved. It turned out that one reason for the problem is the lack of qualified instructors: It might sometimes be necessary to move the teaching given by them because of the collisions with their own studies. Due to all cascading effects it takes time before the teaching program gets stable. Moreover, some of these problems appear very late in the process since a suitable instructor is found so late.

2. Laitoksella opiskelijoiden ja opettajien välinen vuorovaikutus on useimmiten helppoa ja luontevaa

When the department was smaller, the interaction between the students and teachers was easy, but in a larger department communication can be a problem. Actually, according to the students there are problems in information services. The background of many decisions is not passed on to the students. Department's web pages were also criticized even though I found them reasonable, but they could be better. Any way, some strategic plans for improving information services seem to be necessary.

3. Laitoksella on meneillään opetuksen kehittämishanke, jossa selvitetään opintokokonaisuuksien nykyinen tilanne ja pyritään löytämään ratkaisut mahdollisiin ongelmakohtiin.

All the difficult questions were transferred to the teaching-improvement work group. I expected some strategic planning already in this phase knowing that the teaching-improvement work group is basically the same as the self-evaluation work group.

4. Koulutuksen sisällölliseen kehittämiseen tietojenkäsittelytieteen laitos pyrkii seuraamalla alan koulutusohjelmien kehitystä mm. ACM:n Computing Curricula 2001 avulla.

I expected a detailed comparison between the current curriculum and ACM's Computing Curricula proposal[3] from December 2001, and some strategic decisions if there were big differences. It turned out that only two members of the work group had read ACM's proposal making any development work pointless.

5. Koulutuksen arvioinnin ja kehittämisen yhteydessä mainittu laitoksen opetuksen kehittämistyöryhmä konkretisoi ja priorisoi suunnitelmat aikanaan.

[3] The Joint Task Force on Computing Curricula, *Computing Curricula 2001 Computer Science — Final Report*, Association for Computing Machinery (2001). Available at http://www.computer.org/education/cc2001/index.htm.

The discussion about the future perspectives and visions was avoided and moved to the future — if done at all.

Recommendation 2 *Take up the difficult questions mentioned above in the next phase of this evaluation project. If one is not willing to see the problems, one cannot solve them.*

Organization

The department of computer science has four study lines:

- theory of computing
- computer science education
- software engineering
- signal processing

The curriculum planning is distributed for the professors leading the study lines and for the amanuensis, and the proposals are ratified by the institute board. According to the students the amanuensis is extremely overloaded at the moment. Moreover, I am not sure whether the institute board is the right forum for curriculum matters.

Recommendation 3 *Consider whether there should be a study board that has full responsibility for curriculum planning. For instance, one student from each year's course could be a member of such board.*

The head of the department leads the daily administration. At the moment he is the only person who has the responsibility for economical matters. In my opinion the economical planning is done wisely. The main problem — as in all other computing departments in Finland — is that more than 3/4 of all funding is external and time-limited. It is important for the department to spread the valuable information about the economy for more than one person.

Recommendation 4 *Share the information about economical matters between several people.*

Plans and visions

One needs to understand not only the department's business today but also the direction it is going and its vision over both the short and long term. The strategic planning document[4] available at the web was three years old. In the self-evaluation report two plans were mentioned:

1. The co-operation with the other universities in Northern and Eastern Finland about distance education will be continued.

[4] Tietojenkäsittelytieteen laitos, *Strategia 9.8.1999*. Available at http://cs.joensuu.fi/pages/amanuenssi/strategy.htm.

2. The virtual university courses offered to the gymnasium students will also be offered to the students studying in the campus of Savonlinna.

Both of these initiatives are interesting and I encourage you to go on with these plans.

I expected to see more complete and visionary plans. As to this I expressed my concerns to the head of the department immediately after reading the self-evaluation report.

Recommendation 5 *Put the plans and visions on paper in greater detail so that they could be debated in the department.*

Curriculum

In the basic form the study structure is clear. One ***study week*** corresponds to one week of full time study and it is expected that each student works 40 weeks per year. The master's program takes four years, i.e., 240 study weeks. One should collect one ***laudatur module*** (75 study weeks), and one ***cum laude module*** (35 study weeks) or three ***approbatur modules*** (á 15 study weeks), plus other freely selected studies.

It was difficult for me — and also for the students — to understand why the study lines maintain their own study requirements with arbitrary, small discrepancies. Especially, I find the maintenance of the list, explaining in which form a course is accepted for each single study line, a nightmare.

Recommendation 6 *Simplify the study requirements.*

For instance, each study line could maintain a list of compulsory courses. In connection with each course its prerequisites could be listed, and these should be followed. Hereafter it could be left for each individual student to decide whether she/he wants a deep or a broad education.

I also find the connection between the research groups and study requirements unlucky. Accept the fact that the department is educating some sort of software developers. In my opinion, for instance, the existence of the signal processing study line should be reconsidered since it attracts so few students. This does not mean that the research done in this area is insignificant. Even after a possible discontinuation the current courses could be offered as long as there are students wanting them and having the required prerequisites.

We discussed — as always — the first programming language in the curriculum. According to the web-survey answers the students could be divided into two groups: those who think that Pascal, which is currently in use, is okay and those who think that Pascal is not okay. I agree with the latter group of students.

Recommendation 7 *Change the first programming language in the curriculum to a language which is used in real-world software development. This way the students get used to modern libraries and programming environments available for such a language.*

This proposal is controversial; for alternative paths to follow, see ACM's curricula proposal.

To improve the mobility of the students, I recommend that you immediately move from the study week system to the European Credit Transfer System (ECTS)[5].

Recommendation 8 *Move to ECTS points: 1 study week is 1.5 ECTS, i.e., one study year is 60 ECTS.*

A bigger reform, which has to be done, is to adapt to the Bologna model where B. Sc. education takes 3 years (180 ECTS), M. Sc. education 2 additional years (300 ECTS), and Ph. D. education 3 additional years. Instead of the 3+2+3 model, I have seen in Finnish newspapers recommendations on a 3+2+4 model, but I consider this model harmful. Finnish Ph. D.'s should not be put in an unequal position compared to other European Ph. D.'s. Three-years Ph. D. education has worked nicely in Copenhagen for years, although for talented students a 4 + 4 model without a M. Sc. thesis has been applied with success in Århus.

Recommendation 9 *Move to the* 3 + 2 + 3 *model ratified in Bologna as soon as possible.*

Students

At the moment the department has five kinds of students:

- computer science majors
- computer science minors
- gymnasium students taking the virtual approbatur module
- so-called ***transfer students*** (Most of these students have a degree in some other subject and they will become experts in computing.)
- so-called ***IMPIT-students*** (IMPIT is an international student exchange program. Most of the current students are from Russia or Czech Republic.)

To recruit new computer science majors the department has rented an old library bus and made a tour in the gymnasiums of the region. The idea has been to get the local candidates to select their closest university, instead of moving to bigger university cities like Helsinki, Tampere, and Turku.

A general problem in the whole faculty is that the amount of gymnasium students reading "long" mathematics is lower than the number of open study positions at the natural-science faculties. The department of computer science has solved the problem by requiring only "short" mathematics as a prerequisite. Also, the secondary subject can be selected freely, which attracts students from other regions.

For those students having the "short" mathematics background a separate course for updating their gymnasium mathematics is offered. The same solution is used in many Swedish universities.

Normally it takes one year to pass the approbatur module but in the virtual form the study time is one and a half years. Of the 80 students, who entered the first course,

[5] For more details, see *The ECTS users' guide*, European Commission (1998). Available at http://europa.eu.int/comm/education/socrates/guide-en.doc.

20 passed the course last December. If the students pass the course at least with the grade “good”, they can start their computer science studies directly from the second year.

The transfer students seem to have difficulties in following the courses, especially the ones that are mathematically demanding. I was told that once it was necessary to give a course credit for the students that followed 120 hours lectures even if they could not pass the final examination. To improve the commitment of these students the department requires that they study full time (only half time in Kuopio), offers only time-limited rights to study, and pursues that the preset goals (B. Sc. degree, laudatur module, or M. Sc. degree) are met. It has happened that, when the goals were not met, the right to further studies was refused.

The IMPIT-students are selected carefully. The department has contacts with the local teachers and each student is required to have a recommendation from a contact teacher, and after this only the best candidates are selected. The program has been a success. These good students push even the Finnish students forward. After the graduation many of the students have stayed in Finland, which was not expected.

Recommendation 10 *Try to get permanent funding for the IMPIT exchange program.*

The dropout of the students is considerable, especially during the first year, but this is a common phenomenon in all Nordic countries. It was a surprise for me that so many of the students use the computer science program as a stepping stone for other programs. I find this situation unsatisfactory but for the moment I have no concrete proposal how to handle it.

To encourage the students to finish their studies fast, the department offers Fast-Master stipends for the first five students from each year’s course earning their M. Sc. degree in less than five and a half years. The fastest students have earned the degree in three and a half years.

Teachers

In the department there are open positions that has not been filled due to the lack of qualified applicants. This overloads the current teachers. In my opinion the teaching load of the lecturers is too high. The same is true even for some professors. A university teacher must have time to renew herself/himself occupationally.

Recommendation 11 *Reduce the teaching load of the lecturers, e.g., in the way done at the University of Helsinki.*

I was glad to learn that compulsory courses are solidarily divided for the teachers so that in every second term one gives a compulsory course and every second term a course according to her/his own research interests.

In the discussions with the teachers it became clear that the new total-working-time system (kokonaistyöaikajärjestelmä) does not work optimally. It seems to be unclear for many how the new system should be followed. Rules[6] that are identical for the whole faculty would be clarify the situation. I do not like such rules but they make the teachers equal.

[6] For an algorithm how different activities are honoured in our department, see http://www.nat.ku.dk/sekretariatet/oversigt_1.htm.

Recommendation 12 *Clarify the rules how different teaching activities are honoured in hours.*

The amount of hours honoured should be followed afterwards and a teacher who has, e.g., supervised exceptionally many M. Sc. theses or Ph. D. theses could be freed from lecturing for a term or two, or even offered a sabbatical year.

Teaching forms

Using the term from the Århus dropout study[7] one could say that the education offered by the department is traditional. It is based on lectures, home exercises, and some project work, and it is expected that the students work hard — alone. My assumption is that most master's students become some sort of software developers, and software development as it is today is in a large extend a team sport. When visiting Nokia in Salo three years ago, it was said that they need people who can collaborate — the other qualifications are not so important. I think there is some truth in this statement.

Recommendation 13 *Increase the number of compulsory projects that are to be accomplished in groups.*

Recommendation 14 *Consider whether a M. Sc. thesis can be written jointly by a group of two or more students.*

In my opinion the education relies too much on lectures. For example, a course on data structures and algorithms would include only about half as many lectures in Copenhagen compared to the course with the same content offered in Joensuu. Let the students study, possibly teach each others in study groups, and make them in this way more independent.

Recommendation 15 *Reduce the amount of lectures associated with the courses.*

Some of the teachers have already given experimental courses along these lines. I encourage you to continue these experiments.

In this spring a distance education course on computer graphics is being offered by the University of Oulu via the Internet. In Joensuu the room, where this course could be interactively followed, had seats for around 10 students. In the student evaluation the quality of the video-conferencing equipment was criticized, so the equipment should be modernized if this teaching form will be used in the future.

[7] Ulrik Larsen, *Frafald og studiemiljø*, Studenterrådet ved Århus Universitet (2000). Available at http://sr.au.dk/frafald.htm. A review of this study in English, written by my student Christopher Curry, is available at http://www.diku.dk/research-groups/performance-engineering/BPR/SRAssessment.pdf.

Teaching material

In the course descriptions in connection with the study requirements the textbooks used are rarely mentioned. I got the impression that, in the courses I know well, the best books are not used. At least in the courses on algorithmics, computer architecture, and operating system concepts a change of the textbook might be necessary.

I also wondered why so many teachers write their own lecture notes — in Finnish. For well-established courses good textbooks are available. Why not use them? Is, for instance, the Finnish software-development tradition so different from the international tradition that own lecture notes are necessary? Often the books offer reasonable slide material on the associated web pages. By using them, sparse teaching resources could be saved. On the other hand, some of the lecture notes have been published as books for the Finnish audience, which of course is positive.

It is good that instructions for M. Sc. thesis writers are provided. However, the current instructions could be improved at least in the following ways:

1. Instead of listing some random typical mistakes, refer to well-known language guides[8].
2. Instead of giving a collection of formatting rules, provide, e.g., a LaTeX style `jucs-thesis.cls` and a BibTeX style `jucs-thesis.bst` to be used in all theses.
3. Provide also references to books[9] that discuss more severe problems in theses writing: inability to write, conflicts with the supervisor etc.

When looking at the M. Sc. theses exhibited, I found it strange that the statement of the examiners and the grade were published in the beginning of the theses. I think you should remove them and let the reader herself/himself evaluate the thesis. For instance, when I evaluate a thesis, part of the grade comes from the student's working process, possibly from a software that is not documented, and these cannot be seen in the end-product.

Examination forms

I learned that most examinations are traditional written examinations without any support material. Often such an examination only tests how much material a student can push into her/his short term memory, not how well she/he understands the subject. I have good experiences with oral examinations and I recommend them warmly.

Recommendation 16 *Try different kinds of examination forms, for example, oral examinations.*

Recommendation 17 *Allow the students to have their textbook and notes with them in some of the traditional written examinations.*

[8] For instance, Terho Itkonen and Sari Maamies, *Uusi kieliopas*, Tammi (2000) or William Strunk Jr. and E. B. White, *The Elements of Style*, 3rd Edition, Allyn and Bacon (1979).

[9] For the literature I recommend for my M. Sc. students, see http://www.diku.dk/undervisning/2002f/741/.

I heard that in the computer-education group several other examination forms have been tried: portfolio examination, learning diaries, assessment of home exercises, creation of a poster, and writing on an essay. These could also be tried in other courses when applicable.

The Finnish tradition seems to be small courses and small examinations. A single course can have two intermediate examinations, one in the middle and one at the end. Hereafter, if a student has not passed both of these intermediate examinations, she/he still has several chances to pass the course through a normal examination.

Recommendation 18 *Consider whether it is possible to reduce the number of examinations.*

For example, in Copenhagen in the first-year course Dat0[10] (20 ECTS) there are three project examinations and one written examination (with textbooks and notes) at the end. I make no claims that this is good; take this only as an example. Each student has three chances to pass the written examination — once in summer, once in winter, and once in next summer together with the next generation of students. If one passes an examination, it is not allowed to improve the grade. If the student cannot pass these first-year examinations, she/he has not right to continue her/his studies.

Resources

The computer-science building is new and fresh — even if it in my taste is too technology oriented, but I know that technology is loved in Finland. One can even see art on the walls. The offices were a bit crowded so a copying machine was placed on the corridor beside the meeting room we used. It was said to disturb some of the staff members. However, the department will soon get some new offices in the extension of the current building, where after there should not be any space problems.

About 11 students per workstation sound a bit large, but the amount of project work is relatively small at the moment. If the amount of compulsory projects will be increased, the number of workstations should be increased correspondingly. I would wish some more rooms for students working in groups. Now there is only one such room available. Also, the library contained exceptionally few places where one could do home exercises or write theses.

The students asked for a study advisor, which seems to be missing, even if the department offers tutoring for the first-year students and has even five tutor teachers.

Internationalization

The department has used some of their transfer-education funding to recruit foreign IMPIT-students. The first 11 IMPIT-students have just finished their studies with good results. These students have brought welcome international colour to the activities of the department. Otherwise, the student exchange to and from the department has been very limited.

Annually about 40 study weeks (about five courses per term) are offered in English. This has caused three problems:

[10] For further details, see http://www.diku.dk/undervisning/undervisning1.del/1.del.k.beskr.00.01/dat0.htm.

1. In the interviews the students said that they feel that they have the right to get all compulsory courses in Finnish.
2. In their evaluation the students wrote that some teachers do not speak English well enough.
3. The software-engineering group wrote in their self-evaluation that the foreign students cause extra work since separate teaching material must be provided in English.

All these problems are real, but I feel that the last two are only temporary. After the teachers get more practice their ability to use English will automatically get better. To avoid double work more teaching material must be provided in English, and only in English. The first problem is the most severe and it requires some actions:

Recommendation 19 *Declare at the faculty or the university level that both Finnish and English are official teaching languages.*

The coin has two sides: the internationalization of Finnish students is important, but also the foreign students should be able to understand Finnish at least if they study longer than two years.

Industry collaboration

After the move to the science park it has been easy to establish contacts to the firms having their offices in the same building. In these firms many employees have graduated from the department or are writing their M. Sc. thesis. This proximity is already seen to have some synergy effects.

In the software-engineering study line two industrial training periods (7 and 10 study weeks, respectively) are compulsory at the laudatur level. Because of this training the group has tight contacts to some of the local firms. The group has even gone so far that they rank the firms based on the methods used in their software production. Only four of the local firms have received the highest ranking, which has caused some problems for the students to find a training position.

All in all, I got the impression that the collaboration with the industry works well, and because of the ranking system the department even helps the firms to improve their production methods. Apparently, the number of M. Sc. theses done in the industry could even be larger than it is today.

Quality control

Many of the recommendations given so far involve only one time changes. The most important tool for the continuous self-improvement is regular course evaluations. Currently, there exists a web-based course-evaluation system, but very few of the students provide feedback and the summaries generated automatically for the head of the department are not very informative. Furthermore, for the students no response is provided.

Recommendation 20 *Improve the course evaluation system and make the course analyses summarizing the evaluations publicly available for the whole department — including the students.*

To get the students to answer the questionnaires several tricks might be necessary: provide extra bonus to the final grade of a course for a returned evaluation or let a returned evaluation trigger the study credit of a course. Some soft enforcement might also be necessary to get the teachers to write the course analyses based on the evaluations written by the students. My experience is that the students come with many good ideas, but sometimes it is necessary to censor the comments before the publication of the analyses.

Yet another continuous quality control mechanism — which works extremely well in Denmark[11] — is to let a person external to the department function as the second examiner for written projects and M. Sc. theses. Since the number of M. Sc. theses is so small, this system could be applied for them and even a small honorarium could be paid for the external examiners.

I also see the alumni activities — that has just started — a way to get continuous feedback on the quality of the education. I recommend that the department sees their own alumni as a resource.

Ph. D. education

At the moment there are 17 Ph. D. students at the department. The funding comes from the department budget (as assistantships), from the Eastern Finland Ph. D. school, from other external sources like Tekes, and some students finance their studies themselves. Two of the teachers (Pasi Fränti and Jorma Sajaniemi) are supervisors for 6–7 students each. I consider this problematic as one of the teachers said the supervision of a Ph. D. student requires mental resources. I would not supervise more than three Ph. D. students at a time knowing how demanding their supervision is. I hope that all Ph. D. students have a supervisor group consisting of 2–3 teachers so that the work load could be divided more evenly.

In the discussions it also became clear that there is no common practice how the supervision is done. In particular, I am worried about the updates made to the working plans; sometimes these are done, sometimes these are not done at all. The Swedish report recommended that these updates should be done once a year, but I know a firm where similar type of progress discussions are done once every third month. In Turku Centre for Computer Science even the supervisor is required to write a progress report once a year.

Recommendation 21 *Require that the Ph. D. students update their working plans once a year.*

In some departments (Chalmers and Linköping) in connection with such updates an internal workshop is organized where the Ph. D. students present their results for other staff members.

Now afterwards when reading the study requirements for a Ph. D. degree I feel that the requirements could be more specific than “special studies in computer science”. I assume that this means participations in summer schools, talks in international conferences etc.; this could be made explicit. For the sake of internationalization, I recommend that you change the study requirements as follows:

[11] For more information about the Danish censorship system, see, e.g., http://www.retsinfo.dk/_GETDOCI_/ACCN/B19930033205-REGL.

Recommendation 22 *Make a half year or longer visit to a foreign university or research institute a compulsory part of the Ph. D. studies.*

Final remarks

Our visit was shadowed by a conflict, which was experienced a few weeks earlier and where most of the professors were involved. Such overheated discussions are not at all unusual in larger departments, and e-mail is not the right forum for them. I have not made any attempts to analyse the happening in question; I only recommend that the department — or the faculty — tries to find measures to handle any forthcoming conflicts. Already in this point it might be a good idea to elect one official for taking care of such conflicts. It is best to handle such conflicts as soon as possible before they get out of hands and do damage to the whole working environment.

As we emphasized during our visit, the oral feedback we gave was an important part of the evaluation. Immediately after the visit I felt that we could stimulate the thoughts of individual staff members. I hope that this written report keeps the process going, and that the heavy everyday duties many of the staff members have do not stop the process.

I wish you good luck when making the strategic decisions.

Copenhagen, 31.3.2002

Jyrki Katajainen

Appendix 1: The department in numbers

Department of Computer Science, University of Joensuu

Offered degrees

B. Sc. (120 study weeks)
M. Sc. (160 study weeks)
Licentiate in Philosophy (200 study weeks)
Ph. D. (200 study weeks)

Study lines

theory of computing
computer science education (since 1992)
software engineering (since 1996)
signal processing (since 1998)

The number of students

Intake: positions/started
1999:
70/73 computer science majors
25/29 transfer students
2000:
80/70 computer science majors
45/42 transfer students
2001:
90/84 computer science majors
45/37 transfer students

Intake to different study lines after the first year
1999:
8 theory of computing study line
? computer science education study line
14 software engineering study line
1 signal processing study line
2000:
8 theory of computing study line
8 computer science education study line
25 software engineering study line
1 signal processing study line
2001:
10 theory of computing study line
10 computer science education study line
15 software engineering study line
1 signal processing study line

The number of M. Sc. thesis writers
1999:
12 theory of computing study line
2 computer science education study line
0 software engineering study line
0 signal processing study line
2000:
8 theory of computing study line
1 computer science education study line
1 software engineering study line
0 signal processing study line
2001:
12 theory of computing study line
2 computer science education study line
7 software engineering study line
3 signal processing study line

The number of Ph. D. students
1999:
8 in all

2 new
2000:
14 in all
7 new
2001:
17 in all
4 new

The number of teachers

In total: 18, of which 10 have earned a Ph. D. degree
Professors: 5, of which 1 has a time-limited contract
Lecturers: 3
Senior assistants: 4
Assistants: 6, of which 4 have a short-term contract
Instructors: ?

The amount of teaching and research obligations

These figures are rough estimates and are based on one person only.

Professors: 35% teaching, 50% research, 15% administration (Pasi Fränti)
Lecturers: 75% teaching, 10% research, 15% administration (Simo Juvaste)
Senior assistants: 50% teaching, 35% research, 15% administration (Markku Hauta-Kasari)
Assistants: 25% teaching, 75% research (Ismo Kärkkäinen)
Ph. D. students: ?
Instructors: ?

The number of women

Teachers: circa 21%
Computer science majors: circa 25%
Ph. D. students: circa 12%

International student exchange

Foreign students in Joensuu
1999: 2
2000: 2
2001: 2

Own students in a foreign university
1999: 3
2000: 2
2001: 2

Foreign teachers in Joensuu (in teaching duties)
1999: 0
2000: 0
2001: 1

Own teachers in a foreign university (in teaching duties)
1999: 0
2000: 0
2001: 3

The number of students per workstation

The number of students per workstation: circa 11
Access: 24 hours per day

The amount of guided teaching

First year: circa 12 hours per study week

Supervisor capacity in the Ph. D. education

In total: 16
Professors: 5
Docents: 6
Lecturers: 3
Senior assistants: 2
The number of Ph. D. students per supervisor: circa 1

The amount of courses in further education

Ph. D.: courses 40 study weeks, thesis 0 study weeks
Licentiate in Philosophy: courses 40 study weeks, thesis 0 study weeks

Participation of the Ph. D. students in international conferences

1999:
0 talks
0 participations but no talk
2000:
9 talks
0 participations but no talk
2001:
? talks
? participations but no talk

The amount of study credits earned

Approbatur modules taken
1999:
23 in all (education study line missing)
2000:
42 in all
2001:
26 in all (education study line missing)

Laudatur modules taken
1999:
17 in all
13 theory of computing study line
3 computer science education study line
1 software engineering study line
0 signal processing study line
2000:
15 in all
12 theory of computing study line
2 computer science educations study line
1 software engineering study line
0 signal processing study line
2001:
27 in all
11 theory of computing study line
1 computer science education study line
12 software engineering study line
3 signal processing study line

The number of degrees

M. Sc. degrees
1999:
17 in all
12 theory of computing study line
4 computer science education study line
1 software engineering study line
0 signal processing study line
2000:
9 in all
8 theory of computing study line
1 computer science education study line
0 software engineering study line
0 signal processing study line
2001:
20 in all
12 theory of computing study line
2 computer science education study line
6 software engineering study line
0 signal processing study line

Ph. D. degrees
1999: 0
2000: 1
2001: 1