High School Students Learning University Level Computer Science on the Web – a Case Study of the *DASK*-Model

Linda Grandell
Turku Centre for Computer Science
Åbo Akademi University, Turku, Finland

linda.grandell@abo.fi

Executive Summary

Computer science is becoming increasingly important in our society. Meta skills, such as problem solving and logical and algorithmic thinking, are emphasized in every field, not only in the natural sciences. Still, largely due to gaps in tuition, common misunderstandings exist about the true nature of computer science. These are especially problematic for high school students, who need to have a realistic view of what studying computer science would be like in order to make advised decisions about their future careers. The principal objective of Finnish high schools is to prepare young students for further studies and life in general; it can therefore be seen as something of a paradox that these schools are not obliged to provide their students with tuition in computer science.

In this paper we describe the project *DASK* (DAtakunskap i SKolan, Computer Science in Schools), which aims at introducing computer science to Swedish-speaking high school students by providing university level computer science courses on the web. The purpose is to give a realistic view of computer science, at the same time removing any potential misunderstandings about its true nature. The *DASK*-curriculum includes web-based versions of the five basic courses taught at the university with no trade-offs in either contents or difficulty level. The main focus is on student activity, emphasizing the importance of continuous work throughout the courses. We have formulated the *ActiWe* (*Acti*ve on the *We*b) principle, a main guideline applied when designing, developing and giving the courses. Instructors at the university are responsible for the courses in most ways (acting as the course lecturer, maintaining the material, checking assignments, giving feedback etc.), but each high school has a contact teacher who supports the students locally.

DASK was initiated in 2002, and has been popular from the very beginning; feedback from both students and teachers has been positive. In a recently conducted survey (January 2005), many of the teachers stated that their schools would not be able to provide any computer science-related courses on their own, thus indicating a definite need for the *DASK*-courses. In addition, the high number of participants registering for the first course each year implies that there is a genuine interest for computer science among high school students.

Editor: Lynn Hunt

Our experience shows that the greatest challenges for both students and university instructors are time-related: in most schools, the *DASK*-courses are not given time in the schedule and are thus not part of the actual tuition; this leads to already stressful periods becoming even more burdensome for the students. Giving the *DASK*-courses has also proved to be time-consuming for the university instructors; much time and effort has to be invested in maintaining the material, but particularly in giving individual feedback for assignments to all students on a weekly basis. A future challenge is thus to improve the time management of both students and instructors.

Key words: computer science, secondary education, web-based education, university level courses, distance learning

Background

Today's society requires citizens to have a greater knowledge of information technology. Strategy programs at a national level define the skills needed in an information society and encourage people to learn these skills (Ministry of Education, 2004). In order for everybody to have the opportunity to learn the basics of the new technology, computer science (CS) should be included in general education.

The Finnish Educational System

In the Finnish educational system, high schools (HSs) are referred to as upper secondary schools. The principal objective of these schools is to provide students aged 16-19 with general, all-round education preparing them for further studies and life in general.

According to Ben-Ari (2004), a science can be taught at lower levels when it matures. CS has traditionally been taught mainly at university level, but Ben-Ari argues that the time has come for it to be introduced at lower levels of education as well. For instance, in the USA (Merritt et al., 1993) and in Israel (Gal-Ezer, Beeri, Harel, & Yehudai, 1995), CS curricula for HSs were developed in the 1990s. In Finland, the situation seems to be going in the opposite direction: CS is not part of the HS core curriculum (National Board of Education, 2003). CS has thus a completely different position than other sciences, such as mathematics and physics, which are compulsory at secondary level.

Considering (1) the objective of HSs, (2) Ben-Ari's vision and (3) Finland being one of the leading nations in the field of information technology, it can be seen as something of a paradox that CS is not included in the core curriculum of HSs; institutions that should provide all-round learning are not obliged to provide their students with tuition in perhaps one of the most important subject areas today.

Implications of the Lack of CS in High Schools

When there are no formal requirements for instruction in CS, many HSs are not likely to expand the range of courses to include genuine CS courses; in the worst case, they will arrange no CS-related courses at all. If CS-related courses are taught at all, they are largely practical, aimed at teaching the students how to use the computer and various software packages. The essential skills needed by all citizens in everyday life, such as algorithmic and logical thinking skills, are not developed when simply using the computer as a tool.

Moreover, misconceptions of CS as focusing on computer hardware are reinforced when instruction is concentrated on purely technical and practical aspects. This has implications both for the general public and for future computer scientists; in particular, female students may be intimidated by the stereotypical, nevertheless common, view of CS represented by a young male hacker sitting in front of his computer drinking soft drinks and eating pizza (Martin, 2004).

These misconceptions may also result in students enrolling in CS courses on the misconception that they will be of a practical nature, being unaware of the distinction between computer literacy and CS. During ITiCSE 2003 (Innovation and Technology in Computer Science Education - an international conference for researchers and practitioners in the field of CS education), a working group convened around the topic "Preparing students for university study in CS." Among other things, it analyzed the performance of CS students in different countries during their first years at university (Alexander et al., 2003). One conclusion was that students do not know what to expect when enrolling for studies in CS: “students who join a degree programme on the basis of experience in using PCs and games consoles are likely to face something of a culture shock” (p.144).

Furthermore, researchers have studied factors predicting success in university CS, and many of them have found that pre-university CS courses increase the chance of students succeeding at university level (Alexander et al. 2003; Hagan & Markham, 2000; Taylor & Mounfield, 1989). CS education at secondary level has an important role preparing the students for the university CS curriculum: not only to give students basic CS skills, but also to make them familiar with the genuine nature of the field. How are they supposed to be able to know if CS is something for them, if they do not know what studying it means in practice? Giving HS students a more accurate image of CS might also increase girls' interest in the field, since the gender division seems to appear already at that level (Paakki, 2003). As long as Finnish HSs are not obliged to provide their students with CS tuition, other measures are needed in order to guarantee that HS students are given the opportunity to acquire the knowledge and skills they need.

Web-Based Solutions

Web-based education (e-learning) has been used to improve accessibility of courses at higher levels of education for a relatively long time (Lawhead et al., 1997; Vetter & Severance, 1997). Today, we are experiencing an increasing introduction of web-based courses at lower educational levels, e.g. in Finnish high schools, via projects such as “Etälukio” (Distance high school) and “Virtuaalikoulu” (Virtual school) (“Etälukio” 2005; “Virtuaalikoulun verkko-opintotarjonta”, 2005). These projects offer a new type of service, widening the range of courses available to high school students, by making it possible for them to complete courses at other schools using the web. The “Etälukio” portal offers courses in twenty different areas; none, however, in CS. Similarly, no CS-related courses could be found when searching the “Virtuaalikoulu” course database.

Starting in 2000, the University of Joensuu in Eastern Finland began offering a web-based university level curriculum covering introductory CS to HS students in the surrounding region within a project funded by the Finnish Ministry of Education. Today the *ViSCoS* (Virtual Studies of Computer Science) program includes eight courses making up a total of 22.5 ECTS credit points (“ViSCoS”, 2005).

While *ViSCoS* made CS courses available to Finnish-speaking HS students in the Eastern parts of the country, Swedish-speaking students still lacked the same opportunity. Åbo Akademi University is the only Swedish-speaking university in Finland, and is thus responsible for providing education to the Swedish-speaking minority, which in 2004 made up 5.5 % of the total population (Statistics Finland, 2005). Offering instruction to Swedish-speaking students is, however, quite a challenge, since they are geographically dispersed over a large area, rendering it impossible to think that the university could provide students with face-to-face instruction. It would not be reasonable to expect students to come to the university regularly, or teachers from the university to travel around the country in order to give lectures. One, and perhaps the only, plausible solution is thus to utilize the web for delivery of instruction.

Against this background, an initiative to provide HS students with opportunities to acquire solid knowledge in CS was taken at the Department of Computer Science at Åbo Akademi University in 2002. Using the Internet we have been able to provide university level instruction to HS students more than 400 kilometers apart from Helsinki in the South to Nykarleby in the North.

We shall begin by describing the *DASK*-model, i.e. the web-based CS program. A high level of student activity characterizes all DASK-courses, and is the focus and primary goal of the *ActiWe*-principle. *ActiWe* has served - and continues to serve - as the main guideline when developing and giving the courses. Thereafter we shall present results from surveys conducted among participating students and teachers, as well as discuss our own experiences from the project so far. We conclude the paper by summarizing our experiences and presenting some future challenges.

The *DASK*-Model

The main goal of *DASK* (DAtakunskap i SKolan, CS in schools) is to introduce CS to HS students by providing university CS courses in Swedish on the web. The purpose is to give the participants a realistic view of what it would be like to study CS at university level, and in this way diminish the gap between HSs and the university. Establishing contacts with the schools, their teachers and students, makes it possible to market the department in a new way and to develop a close co-operation with teachers in mathematics, physics, chemistry and other natural sciences. It is important to pay special attention to the relationship to the teachers, since they play a big part in the students' choice of further studies.

Each *DASK*-course counts as two HS courses. In addition to this immediate benefit, the students also receive other advantages: they are accredited for each completed *DASK*-course at Åbo Akademi University, and do not have to retake the courses corresponding to these if/when they enroll for further studies at the university. Moreover, HS students who have completed 15 study points (sp) of the *DASK*-curriculum with good results are accepted as students at the department without having to take the entrance exam.

DASK was thus originally a co-operation project between our university department and Swedish-speaking HSs around the country. Since 2004 other interested persons may, however, also participate in the courses via the Open University (Öppna Universitetet). This makes it possible for working adults and others to complete courses in CS without having to attend any lectures on a regular basis. This has also turned out to be a welcome alternative for university students who are not able to attend day-time lectures due to a full schedule, overlapping lectures or working hours; by participating in the web-based versions they can still take the courses, studying when they have time.

The department does not receive any external funding, but provides the *DASK*-courses free of charge to high school students. The only compensation in form of money comes from the Open University, and the amount is based on the number of students registering to the courses.

ActiWe - Active on the Web

One cannot learn CS by only reading a book or listening to lectures; students trying to understand concepts by only going through rules tend to waste both time and effort. In order to learn, they need to be active, solving exercises, going through examples, analyzing problems and so on. Although this might be considered something of a matter of course, we think it is particularly essential to keep in mind when designing web-based courses.

Based on the principles of *active learning* ("Active", 2005; Bonwell & Eison, 1991), we have formulated the *ActiWe*-model (*Acti*ve on the *We*b) in order to design suitable, web-based material covering the prerequisite course contents. *ActiWe* also follows a *constructivist* approach (Ben-Ari, 1998; Jonassen, Peck, & Wilson, 1999), taking focus away from the teachers and making the courses learner-centered. The main aim of *ActiWe* is to engage the students in their own learning by having an activity-intense syllabus, including different types of activities in order to cater for different learning styles. Each course still includes a final exam, but we have de-emphasized its importance and instead highlighted the significance of the continuous work done throughout the course.

John Dewey's (1910) principle of "learning by doing" is at the core of all courses. The students are to submit assignments on a weekly basis, and the points scored in these play a substantial role for the final grade in addition to the exam. Furthermore, each course includes numerous optional activities, such as animations, exercises, traditional and interactive examples and laborations (laborations are a new type of exercises, in which the students are given a problem and are led through its solution step-by-step with clear instructions on what should be accomplished in each step and why). As an example, Figure 1 shows a screenshot from an interactive animation illustrating how a decimal number is converted into a binary one. The students can work with the optional activities on their own, if and when they want to.

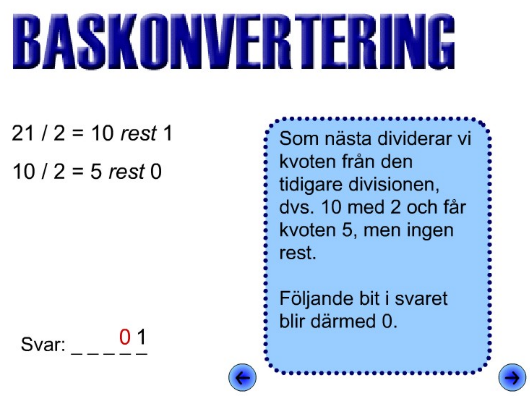

Figure 1: Interactive animation illustrating decimal-to-binary conversion.

Good web-based courses are not created by simply converting traditional course material into electronic form. We have tried to make use of the many benefits of e-learning but also pay attention to its potential drawbacks (Kruse, 2004; Zhang, Zhao, Zhou, & Nunamaker, 2004). To minimize the risks of students feeling isolated, the instructors keep in touch with the students on a regular basis and give individual feedback on assignments. Moreover, we have aimed at only including technology that is easy to use, also for novice computer users. In addition to these issues, we have paid much attention to the fact that the *DASK*-courses are primarily aimed at students on a lower level of education than the corresponding courses were originally intended for.

Curriculum

The main difference between *ViSCoS* and *DASK* is that no new curriculum was created for *DASK*. *DASK* includes five courses (listed in Table 1), and these are the same as the basic courses offered by the department. We had several reasons for taking this approach. First, the same material can be used in traditional tuition as well. Second, the courses being web-based versions of the original ones makes it possible for students to get an accurate picture of what studying CS at university level is like, thus improving their understanding of CS as a field of scientific study. Finally, this was an academic necessity since the *DASK*-courses must correspond exactly to the versions at the department in order for these to be interchangeable.

Table 1: **Courses included in the *DASK*-curriculum**

Introductory Course in CS (*Intro*)	5 sp
Programming, Basic Course (*ProgBC*)	5 sp
Programming, Continuation Course (*ProgCC*)	5 sp
Computer Hardware Basics (*CompHB*)	5 sp
Logic	5 sp

The first three *DASK*-courses have to be taken in the order listed in Table 1, given that each course builds on knowledge acquired in the previous ones. Since these are also the courses that are likely to attract most participants, they are given annually, whereas *CompHB* and *Logic* are offered every second year. The schedule has been planned to make it possible for students to complete the entire curriculum in at most three years.

Course Delivery

All course material is made available on the web using the course management system *Moodle* (http://www.moodle.org). Moodle is an open source platform well suited for our purposes, being flexible, customizable and featuring many different types of activity modules.

The screenshot in Figure 2 illustrates the main page for one of the courses. The design of web-based courses should follow the same guidelines as other "traditional" web pages, and a consistent look-and-feel is thus a key factor in making the courses accessible and easy to use (Opper-

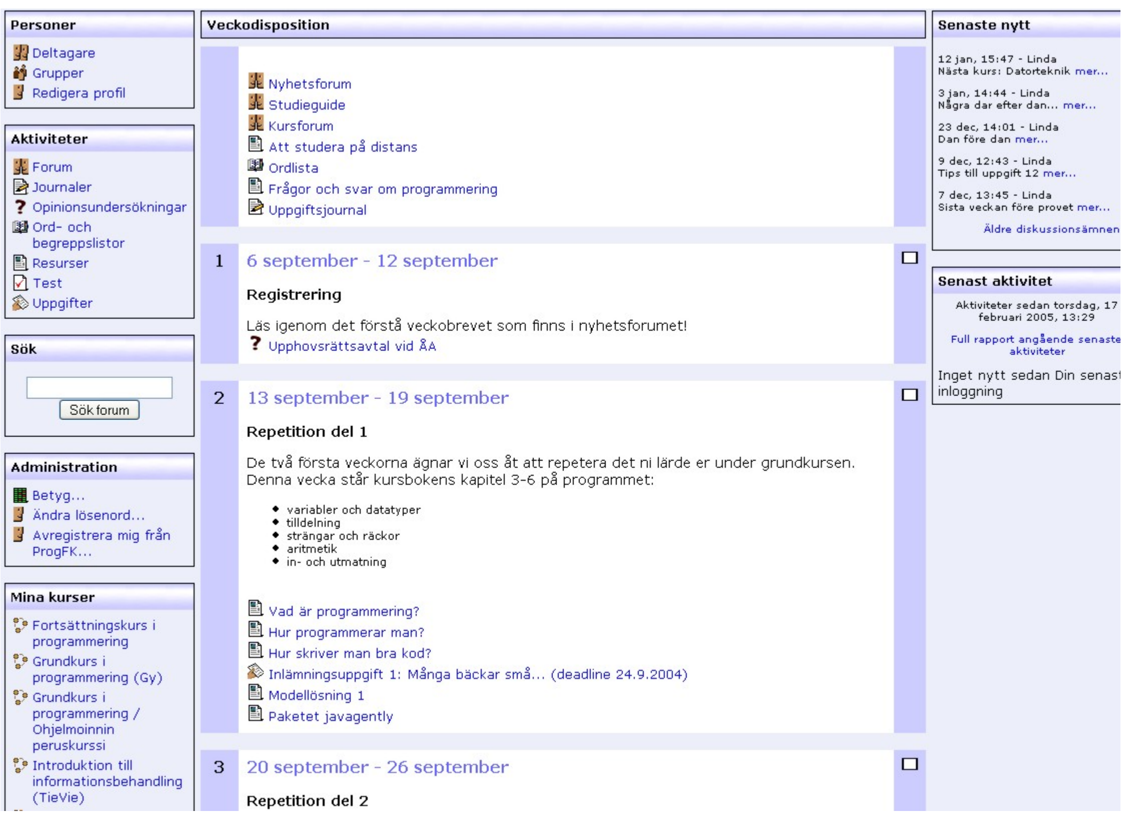

Figure 2: The main course page for one of the *DASK*-courses

mann, 2002; W3C, 1999). Using the same system for all courses gives them a similar layout, and by using the same type of material and activities in all courses we ensure that students know what to expect when beginning a new course. Moodle serves as the classroom and the main communication medium between students and the instructor. The contents for each course are divided into smaller parts, each of which makes up one week in the course schedule. The weeks are illustrated as boxes on the course page, one box containing all instructions, reading material, assignments, visualizations, exercises, examples, and so on related to the topic to be covered during that week. This makes it easy for the students to get a clear picture of what is to be done during a specific week.

Contact Teachers and Instructors

Faculty at each school assigns a contact teacher for the *DASK*-courses. Initially, the idea was that each contact teacher would arrange weekly meetings with the *DASK*-students at his or her school. This has, however, proved to be impossible, largely due to the lack of resources. The contact teachers still play an important role in the courses, mainly as administrators, e.g. taking registrations, making sure the students have access to computers in the school, overseeing exams, and acting as local support for the students.

Each course has a main instructor at the university department, who is responsible for the course in most ways: he or she acts as the course lecturer, posts weekly announcements, checks assignments, gives feedback, moderates discussions, keeps in touch with the students and the contact teachers, answers their questions, prepares and checks examinations and so on. Depending on the number of students, additional assistants may be necessary for helping the instructor, in particular with checking assignments and giving feedback. In the following we will refer to the contact teachers as teachers and to faculty at the department as instructors.

Evaluation

Working with *DASK*, we are to some extent following the principles of *action research* as defined in the book "Computer Science Education Research" (Fincher & Petre, 2004). In action research, practitioners improve their practice by doing or changing something and then reflecting on the results. The main purpose is to collect data and experience that help in gaining a better understanding of the practice.

We reflected on our experiences (what worked and what would benefit from being changed) after each course and made changes and improvements we found necessary. We believe it is essential that the courses are kept up-to-date in order to continue to be motivating for students. Student and teacher feedback, both formative and summative, has been the most valuable source of information for this cycle of making reflections and changes.

Student Surveys

A survey conducted after the very first course in 2002 showed that the students appreciated the course (Grandell & von Wright, 2003). They enjoyed the web-based alternative of studying and the majority stated that the course had helped them develop a clearer understanding of CS. In addition, computer-based aids, which describe and explain different phenomena, were regarded as tools potentially able to assist in developing an understanding of CS concepts. The data derived from later surveys are in line with these results.

A pilot version of *ProgCC*, one of the more advanced courses in the program, was given in fall 2004, and in the post-course survey, the students (n = 8) stated that they thought they had learned a lot about CS by taking the three *DASK*-courses. They were also unanimous in stating that the courses had given them a realistic view of what it means to study CS. In addition, results from the survey and the course activity logs generated in Moodle, showed that the different types of activating material were used to various degrees. However, each type was found useful by some students, which implies that there is indeed a need for many types of activities and study material; what one student finds to be boring and of no use at all may, by somebody else, be experienced as very helpful and of great importance in the learning process.

Teacher Survey

Until recently, we had not conducted any study on the teachers' attitudes towards and experiences from participating in *DASK*. Instead, all feedback has been received from individual teachers at different points of time. In order to obtain comparable and informative feedback from all teachers, we decided to conduct a survey among them in January 2005. The survey was answered by teachers from all schools that participated in the project at that time (n = 6).

The survey was conducted using e-mail, and consisted of 11 open questions aimed at collecting information about the attitudes and experiences of the teachers. The teachers were also asked to estimate the amount of time they invest in being a contact teacher for each course, and to describe what they do during this time. We also wanted to find out whether the teachers thought they could be able to assist more in supervising the courses. In addition, we were interested in getting ideas and suggestions for how the *DASK*-courses could be improved.

Need for *DASK*

All teachers stated that there is a definite need for the *DASK*-courses. The data showed that *DASK* solves two problems faced by Swedish-speaking HSs today: (1) the schools are small, and often lack resources to offer any extra courses, and (2) courses that might be included in the curriculum often have to be left out, due to too small a number of participants.

The survey results showed that over 50 % of the schools would not be able to offer their students any CS-related courses at all, if they did not participate in *DASK*. In schools in which some courses would be given, they would be mainly of practical nature, thus failing to reveal what CS is truly about.

Best practices

Some of the best *DASK*-practices pointed out by the teachers were the following:

- Students have the opportunity to get acquainted with CS.
- *DASK* offers new challenges to those who think they already know everything about their PC.
- Students can work online when they want to and have time.
- The courses are cost-light for the schools.
- The difficulty level of each course stays the same from one year to another. According to some teachers, this is not necessarily the case with ordinary HS courses, in which the difficulty level might be adapted to the knowledge level of the participants.
- The courses have up-to-date contents and are professionally made.

An additional, very interesting point made by one teacher, is that schools participating in *DASK* may use this to attract students from comprehensive school by marketing themselves as schools offering web-based university level CS courses. Two of the schools have even included the courses in their official curriculum.

The teachers were also asked to point out which aspects of the *DASK* program that their students most appreciated; key phrases such as *freedom*, *web-based*, *academic studies* and *something new* characterized most of the answers. In addition, the contact to other like-minded students in other schools, and the up-to-date contents of the courses, were regarded as things appreciated by students.

Experienced challenges

The results from the survey showed that the main difficulties are related to the schools' schedules, a finding that is consistent with our own experiences. In most schools, the *DASK*-courses are not given time in the schedule and are thus not part of the actual tuition. This has at least two drawbacks: first, the teachers might not have time to help the students as much as would be needed. Second, this leads to already stressful periods becoming even more burdensome for the students, when the *DASK*-courses are put on top of the compulsory ones. We have tried to adjust our course schedules and adapt them to the schedules of different schools as well as individual students, but it seems very difficult to reach a perfect mix of the *DASK*-courses and the compulsory ones.

When asked to state what they think students find most difficult, the teachers mentioned for example, managing to keep deadlines, the need for self-discipline and time needed for the *DASK*-courses, especially during periods when the ordinary courses require a lot of work. Lack of close-by assistance was regarded as an additional difficulty in schools where the teachers found that they did not have time to support the students as much as might be needed.

Lessons Learned

DASK has now been running for three years and it is appropriate to reflect on our experiences so far. The high number of students registering for the introductory course each year indicates that there is an interest in studying CS among students at HS level. For instance, the last time the course was given (Fall 2004), we received almost 100 registrations, and most of them came from HS students. Considering that the Swedish-speaking population is a minority in Finland, this number of students from Swedish-speaking schools can be regarded as quite high. Although the majority of the participants are male, the women taking the *DASK*-courses constitute a larger percentage of the total number of participants (almost 25 %) than is the case with the traditional courses at the department.

Our experience has shown that the number of registrations drops when moving on to the following courses, but we do not, however, see this as a problem. The main purpose of *DASK* is, as mentioned earlier, to introduce CS in its true nature. Therefore we regard it as positive, indeed as as important, that large numbers of students want to try it out. If it then turns out that CS is not for them, that is just fine.

When giving a course, we frequently refer back to courses that the students have already completed, as well as forward to upcoming ones. In our experience, students might sometimes have difficulty in seeing the relevance of some topics, for example, algorithms and different number systems, and it is therefore imperative that they can see a connection between both courses and topics. Giving them concrete information about forthcoming courses in which they will have to use specific knowledge makes learning the topics seem more worthwhile.

The main difficulties faced by the instructors at the university when working with the *DASK*-courses have proven to be time-related. Much time and effort has to be invested in maintaining the material, and particularly in giving individual feedback for assignments to all students on a weekly basis. These experiences are in line with the findings of an ITiCSE97 working group (Lawhead et al., 1997): "There may be a perception that creation of Web-based courses will be an easy, one-time effort: ... In fact, constant maintenance, updating, and delivery costs in time and resources will be required" (p. 37). Methods for reducing the workload for the instructor are therefore called for. One obvious solution would be to increase the numbers of instructors and assistants involved in the courses; however, since the department does not receive any funding for the project, there are no extra resources to invest in it.

Conclusion

In this paper, we have presented *DASK*, an initiative to introduce university level CS to geographically dispersed HS students using the web. The courses have been designed to be student-friendly, web-based versions of the courses given at the university without any trade-offs whatsoever in either difficulty level or contents.

Our experience has shown that there is a wide interest in studying genuine CS among HS students. The results from a survey conducted among the teachers show that many of the participating schools are dependant on the *DASK*-project when instruction in CS is concerned and would simply not be able to offer their students any courses in CS otherwise; there is, thus, a definite need for courses such as the ones provided in *DASK*.

So far students from nine HSs have participated in *DASK*, in addition to individual HS students from other schools registering to the courses via the Open University. The project has now been running for three years, and since the experiences have been positive, there is motivation to extend the project to include more schools. This is, however, a question of available resources at the university: the larger the number of participants in a course, the more the time needed for giving feedback, correcting assignments and supporting the students. The burden on the instructors corresponds directly to the number of students, and a future challenge is thus to find new assessment and support methods in order to reduce the currently extensive workload for the instructors. One possibility would be to involve the schoolteachers to a larger extent, but this would require that they have good CS knowledge themselves, which is not always the case today. In addition, one has to remember that one of the main reasons why we provide these courses in the first place is the lack of resources in the schools for arranging CS-courses on their own. Consequently, the schools would most likely be unable to pay the teachers extra for their work with the *DASK*-courses.

In our opinion, one of the responsibilities of universities is to serve society and the community. If it becomes evident that some part of society is unable to offer necessary services, we find it completely justifiable for the university to provide these services. HS students should be guaranteed the opportunity to acquire the knowledge and skills in CS that they want and need. If HSs do not have the resources to arrange CS education on their own, nothing should hinder universities from taking over this responsibility. However, in order for these ideas to be carried out in practice, universities must be provided with the needed resources in the form of both sufficient funding and motivated faculty. As the survey results showed, the *DASK*-courses are experienced as cost-light in the HSs; this is, however, not the case at the university.

References

Active Learning (2005). Wikipedia. Retrieved May 17, 2005 from http://en.wikipedia.org/wiki/Active_learning

Alexander, S., Clark, M., Loose, K., Amillo, J., Daniels, M., Boyle, R., Laxer, C., & Shinners-Kennedy, D. (2003). Case studies in admissions to and early performance in computer science degrees. In *ITiCSE-WGR'03: Working group reports from ITiCSE on Innovation and technology in computer science education*, 137-147. ACM Press.

Ben-Ari, M. (1998). Constructivism in computer science education. *SIGCSE '98: Proceedings of the twenty-ninth SIGCSE technical symposium on Computer science education*, 257-261. New York, NY: ACM Press.

Ben-Ari, M. (2004). Computer science education in high school. *Computer Science Education*, *14* (1), 1–2.

Bonwell, C. C. & Eison, J. A. (1991). Active learning: Creating excitement in the classroom. Retrieved May 17, 2005 from http://www.ericdigests.org/1992-4/active.htm

Dewey, J. (1910). *How we think*. Retrieved June 21, 2005 from http://spartan.ac.brocku.ca/~lward/Dewey/Dewey_1910a/Dewey_1910_toc.html

Etälukio (2005). Retrieved May 17, 2005 from http://www.oph.fi/etalukio

Fincher, S. & Petre, M. (Eds.) (2004). *Computer Science Education Research*. Taylor & Francis.

Gal-Ezer, J., Beeri, C., Harel, D., & Yehudai, A. (1995). A high school program in computer science. *Computer*, *28*(10), 73-80.

Grandell, L. & von Wright, J. (2003). An introductory web-based course to enhance high school students' knowledge of computer science. *Proceedings of the Third Finnish/Baltic Sea Conference on Computer Science Education*. Koli Calling, October, 2003.

Hagan, D. & Markham, S. (2000). Does it help to have some programming experience before beginning a computing degree program? *Proceedings of the 5th annual SIGCSE/SIGCUE ITiCSE conference on Innovation and technology in computer science education*, 25-28. New York, NY: ACM Press.

Jonassen, D., Peck, K. & Wilson, B. (1999). *Learning with technology: A constructivist perspective*. Upper Saddle River, NJ: Merrill.

Kruse, K. (2004). *The benefits and drawbacks of e-learning*. Retrieved May 17, 2005 from http://www.e-learningguru.com/articles/art1_3.htm

Lawhead, P. B., Alpert, E., Bland, C. G., Carswell, L., Cizmar, D., DeWeitt, J., Dumitru, M., Fahraeus, E. R., & Scott, K. (1997). The web and distance learning: What is appropriate and what is not. Report of the ITiCSE '97 working group on the web and distance learning. *SIGCUE Outlook, 25*(4), 27-37.

Martin, C. D. (2004). Draw a computer scientist*. ITiCSE-WGR '04: Working group reports from ITiCSE on Innovation and technology in computer science education*, 11-12. New York, NY.

Merritt, S., Bruen, C. J., East, J. P., Grantham, D., Rice, C., Proulx, V. K., Segal, G., & Wolf, C. (1993). ACM Model High School Computer Science Curriculum. *Communications of the ACM*, *36*(5), 87-90.

Ministry of Education (2004). *Koulutuksen ja tutkimuksen tietoyhteiskuntaohjelma 2004-2006*. Opetusministeriön julkaisuja 2004:12. Yliopistopaino, Helsinki. http://www.minedu.fi/julkaisut/koulutus/2004/opm12/opm12.pdf

National Board of Education (2003). *Lukion opetussuunnitelman perusteet 2003. Nuorille tarkoitetun lukiokoulutuksen opetussuunnitelman perusteet.* Vammalan Kirjapaino Oy, Vammala.

Oppermann, R. (2002). User-interface design. In H.H. Adelsberger, B. Collis, & J.M. Pawlowski (Eds.), *Handbook on Information Technologies for Education and Training* (pp. 233-248). Berlin: Springer.

Paakki, J. (2003). Tietotekniikan koulutus - nousu ryysyistä rikkauteen. *Dimensio*, *3*, 20-22.

Statistics Finland (2005). Retrieved June 27, 2005 from http://www.stat.fi/tup/suoluk/taskue_vaesto.html

Taylor, H. G. & Mounfield, L. C. (1989). The effect of high school computer science, gender, and work on success in college computer science. *Proceedings of the twentieth SIGCSE technical symposium on computer science education*, 195-198. New York, NY: ACM Press.

Vetter, R. J., & Severance, C. (1997, November) Web-based education experiences. *Internet Watch*, 139-141.

Virtuaalikoulun verkko-opintotarjonta (2005). Retrieved May 17, 2005 from http://www.edu.fi/virtuaalikoulu/lot

ViSCoS - Virtual Studies on Computer Science (2005). Department of Computer Science, University of Joensuu. Retrieved April 6, 2005, from http://www.cs.joensuu.fi/viscos/index.php?lang=eng

W3C (1999). Web Content Accessibility Guide 1.0. Retrieved May 17, 2005, from http://www.w3.org/TR/WAI-WEBCONTENT

Zhang, D., Zhao, J. L, Zhou, L & Nunamaker, J. F. Jr. (2004). Can e-learning replace classroom learning? *Communications of the ACM*, *47*(5), 75-79. New York, NY: ACM Press.

Biography

Linda Grandell received her MSc in computer science from the Department of Computer Science at Åbo Akademi University in 2003. She is currently a PhD student at the department and at Turku Centre for Computer Science. She is one of the initiators of the DASK-project and also acts as one of the instructors giving the web-based courses. Her main research interests are related to web-based learning and CS education, and her primary focus is on teaching programming to novices at both secondary and university level.