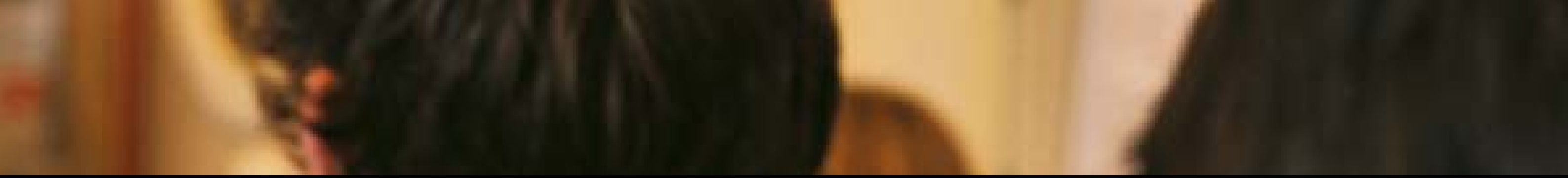
UiO : **Faculty of Mathematics and Natural Sciences**



Computing in Science Education

A guide for universities and colleges in Norway

Working group: Knut Mørken, Nina Sasaki Aanesen, Lars Oswald Dahl, Hugo Lewi Hammer, Terje Brinck Løyning, Anders Malthe-Sørenssen, Elisabeth Nøst, Ingve Simonsen, Jon Eivind Vatne and Tone Skramstad

Date: 1/2/2011. English edition: 15/6/2011

At the request of the Norwegian Ministry of Education and Research (Kunnskapsdepartementet - KD), the working group has developed a model for how computing oriented education can be introduced as an integral part of first degree courses in natural science and engineering.

Graphic production: 07 Gruppen AS

Contents

1 Background

The Norwegian Ministry of Education and Research (Kunnskapsdepartementet - KD) has been monitoring the project, “Computers in Science Education at the University of Oslo”, and refers to the project on p. 36 in the strategy for the natural sciences “Realfag for framtida” (Science Studies for the Future). The strategy for strengthening mathematics and the natural sciences 2010-2014:

> *The CSE project (Computer Science in Education) has been developed at the Faculty of Mathematics and Natural Sciences at the University of Oslo (UiO). The project’s objective is that students shall learn to solve mathematical problems with the help of computers from the very start of their bachelor degree courses. The subject area is called “Numerical Methods”. The problems are realistic exercises, where the students experiment with aspects of the world they live in.*
>
> *UiO is the first university in the world where new students have started working with advanced calculations on computers right from the beginning. Other universities are waiting until much later with introducing students to numerical methods, usually not until master´s degree. At UiO it is seen as pedagogically important to start this interaction from the beginning. In this way the study comes alive from day one, avoiding a long theoretical period in the start phase which for many can reduce their motivation. In addition the students gain an insight into what is happening at the sharp end of research.*

The Ministry (KD) wants to integrate the use of computing at first degree level at more Norwegian higher education institutions. This was the basis on which the mandate for a national working group was prepared.

Name change: The working group is amending the project title to **“Computing in Science Education”** in order to better reflect its mathematical nature, and this is used in the present document.

1.1 The working group’s appointment and mandate

The mandate for the working group was approved by the Ministry on 6th September 2010, and NOK 500,000 was allocated to the Faculty of Mathematics and Natural Sciences at the University of Oslo to carry out the work.

Mandate:

Assignment
The University of Oslo, through the Faculty of Mathematics and Natural Sciences, shall at the request of the Ministry of Education and Research, appoint and lead a working group.

The working group shall prepare a model for how the CSE project (Computers in Science Education) can be adapted to other teaching establishments within MST[1] subjects.

Overall aim

To spread knowledge about the CSE project, with emphasis on the inclusion and integration of a computation perspective in science and technology studies at first degree level.

Delivery

Based on experience gained at the University of Oslo since the launch of the project, the working group shall produce a written guide and contribute to spreading information through activities which underpin the declared aims.

The guide shall as a minimum cover the following points:

- Present the academic argument for emphatically advocating computation in the teaching for first degree students. This to be supplemented with text which can be used by managers as a basis for their decision making process when evaluating which degree courses should have specific teaching objectives for mastering computing oriented mathematics.
- Describe the preconditions, costs and benefits/consequences associated with the introduction of more computing oriented science teaching at an institution. In this context due regard must be given to the diversity in the Norwegian education sector, both regarding resources, size, user groups and subject portfolios.
- Draw up a model for how a computing oriented approach to science subjects can be integrated and implemented in other science and technology education, to include defining the resource and competency requirements.

As far as possible and practical, solutions should be outlined for different levels of specialisation, recommendations for teaching reforms, teaching material etc.

In addition a web site shall be established where amongst other things, learning resources which can serve as good practice can be posted. The working group is encouraged to enter into dialogue with communities in the higher education sector and also with external parties where this is judged beneficial. The working group must itself decide whether any other elements should be included in the assignment, so long as this can be done within the budget.

Composition of the committee

The members of the working group shall, as far as possible, represent the following prioritised criteria:

- different MST communities
- size of the communities
- student representation
- representation from user groups/business
- geographical location
- gender

The working group shall consist of 7-9 people.

Deadlines

- By 1st January 2011: The working group should submit its provisional report to the National Council for Technological Education and the Meeting of the National Science Faculties.

[1] By MST is meant subjects within mathematics, natural sciences and technology.

- By 1st February2011: The working group to deliver its main report to the Ministry (KD).
- By 1st June 2011: To hold a seminar for disseminating information across the sector and to other interested parties.

The working group was selected according to the criteria in the mandate and had the following members:

- Knut Mørken, Professor, Department of Informatics and Centre of Mathematics for Application, University of Oslo. Head of the working group.
- Nina Sasaki Aanesen, student, Norwegian University of Technology and Science, Trondheim – student representative.
- Lars Oswald Dahl, Head of risk analysis, Storebrand insurance group.
- Hugo Lewi Hammer, Associate Professor, Engineering Faculty of Oslo University College.
- Terje Brinck Løyning, Vice Dean at the Technology Section, University College of Narvik.
- Anders Malthe-Sørensen, Professor, Department of Physics and Centre for Physics of Geological Processes, University of Oslo.
- Elisabeth Nøst, Manager – Flow Assurance, FMC Technologies, Oslo area.
- Jon Eivind Vatne, Associate Professor, Institute of Computing and Science, Faculty of Engineering, Bergen University College.
- Tone Skramstad, Adviser, Faculty for Mathematics and Natural Sciences, University of Oslo. Secretary for the working group.

During the period September 2010 to January 2011 the working group has convened for six one-day meetings. These have been held at Gardermoen Park Inn, at Oslo Airport.

The group's leader, Professor Knut Mørken, gave a presentation of the project at the National Science Faculties Meeting on Svalbard 22nd-23rd November, 2010. In the minutes of the meeting it is recorded:

> *The Faculties Meeting thanks Knut Mørken for the briefing and recommends that KD [Ministry of Education and Research] incorporates the CSE model in its strategy for science subjects. The educational institutions should later examine ways in which this can be implemented.*

1.2 Other work in progress

In the period 1st January to 15th October 2010, work has been undertaken to work out a new framework plan for engineering studies in Norway. This was initiated by the Ministry of Education and Research. There is a draft proposal for a new framework plan for engineering studies which will become operative for the academic year 2012/2013 [2].

Members of the working group have been following the progress on this new framework plan, and have themselves responded during the hearing round.

1.3 Further tasks for the working group

As stated in the mandate, the working group was required to arrange a seminar for spreading information about the CSE project. This was held at the University of Oslo on 20th May 2011. ("Fremtidens utdanning: en nasjonal modell for beregningsorientert utdanning")

The working group shall also prepare web pages to be published at www.mn.uio.no/english/cse

[2] The framework plan is posted on: http:// www.regjeringen.no/nb/dep/kd/andre/brev/utvalgte_brev/2011/ny-forskrift-omrammeplan-for-ingeniorut.html?id=632491

2 Introduction

Natural science and engineering are key pillars for the development of society. Energy production, building construction, infrastructure design, materials development, weather forecasting, financial risk management and medical technology are examples of areas which are totally dependent on these fundamentally important subjects.

A common trait in natural science and engineering subjects is that relationships can be expressed mathematically, and this is the key to problem solving in these subjects. The development of computers and software over the last decades has made mathematical problem solving on computers – computing – widely accessible. The result is that problems which previously were impossible to solve or required supercomputers, can now be handled on a standard PC. Vast quantities of data can be processed and visualised in ways that before were difficult to imagine. This revolution has radically changed the subjects and their applications and has led to computing becoming an integral and indeed central part of research and development, and a fundamental tool in business and industry.

Even though the application of science subjects has changed significantly, this is scarcely reflected in the basic education taught at most universities and colleges. In many subjects this means that the students have a limited spectrum of realistic problems that they can relate to in their study, and they therefore have only limited training with important mathematical problem-solving tools. Science education in general will therefore be strengthened if computing is integrated into higher education already from the first semester.

The present document contains observations which can be considered as recommendations both to the Norwegian Ministry of Education and Research (KD) and to educational institutions.

2.1 Recommendations to the Ministry of Education and Research (KD)

- KD should ensure that all relevant educational institutions are made aware of this guide.
- KD should allocate funds which can be shared between the educational institutions to motivate the establishment of a computing perspective in their education programmes.
- KD to establish a national CSE Centre. This will be a resource centre for computing oriented education and will collect teaching material, examples and tasks. The CSE Centre will also initiate and co-ordinate research which will study different aspects of computing oriented education in order to document the results and help establish good teaching practice.

2.2 Recommendations to the higher educational institutions

- All relevant MST degrees should reserve at least five credits for computation-relevant programming training in the first semester, and all mathematical subjects in the first semesters should be given a clear computation perspective which is related to the programming teaching.
- The computing component that is established in the first semesters should be actively utilised in all relevant subjects later in the studies.
- The teaching management should take on the responsibility for ensuring that the computing perspective is preserved and developed, both across subject boundaries and over time.
- The computing perspective should be included in the teaching benefit description for the different courses.

2.3 Contents of the report

In this report we start by explaining why it is important both for business and for science and engineering educations in Norway that the computing perspective is integrated as early as possible in the teaching. We also explain what we mean by computing oriented education and go into considerable detail to show how computing ought to be integrated to achieve good results.

Finally we present three examples of computing oriented education:

1. The University of Oslo where computing oriented education is integrated into first degree courses, how this is done, and experience with the model.
2. How computing oriented education can be integrated at first degree level into both the 3-year and 5-year programmes at NTNU (Trondheim).
3. How computing oriented education can be integrated into the engineering courses at the colleges in Oslo, Bergen and Narvik.

What do "Computing" and "Computing oriented" mean?
Computing oriented problem solving means the solution of a mathematical problem is constructed as a sequence of precise steps that can be executed by a computer. By computing we mean carrying out these steps on a computer.

3 Computing perspectives in business and education

Business and industry demand computing skills
75 % of the Technical Physics students from NTNU are using programming or computing tools in their job one year after graduating.
(Master´s thesis, Arne Stormo, NTNU, 2009)

“The wall failed as a result of a combination of a serious error in the finite element analysis and insufficient anchorage of the reinforcement in a critical zone[3].”

3.1 Computing perspectives in business

Calculations form the basis of many important decisions in the business world, and major investments are often founded on computational results. The quality of the skills employed for computing and analysis are especially critical. But interdisciplinary co-operation also functions better if more participants know the context and details of an analysis and thereby the background for decisions. Decision makers, both before and after computing oriented projects, are definitely able to make better decisions if they have good knowledge of this subject area.

Computing competence is an important requirement both for those who will be carrying out the actual computations and those who will only be using the results.

3.1.1 Cost reduction and quality improvement

In larger projects in business the personnel who combine high qualification in a particular field with a knowledge of computing, will play an important role. Communication and specification account for a significant part of the cost of such projects, and project participants who communicate well across traditional subject boundaries and who understand the possibilities and limitations of computing, are valuable.

Experience with small and medium computing problems, not only using standard software, makes a new employee immediately operative, quickly transformed into a valuable and flexible resource with high productivity: in smaller development projects such candidates will be able to take on a professional role and the results can be permanent software solutions and not just specifications. A high level of computational competence is required to ensure that such solutions do not represent risk factors for errors or instability, or have little distribution potential.

Correct investment in this field will produce savings for business in two ways: increased efficiency and productivity, together with quality gains through an interdisciplinary approach at a more detailed level.

Traditional and modern applications of computations

Computations form the basis of traditional and modern technologies like the development of electronics, computers, advanced materials, ships, aircraft, oil platforms. Computations have the property that they can underpin or replace physical experiments in the laboratory or in the field. Through this, considerable savings and increased accuracy can be achieved. Sometimes computations are the only option when experiments can not be carried out because they are controversial, dangerous, illegal, impossible or expensive. 75 % of the Technical Physics students from NTNU are using programming or computing tools in their job one year after graduating. (Master´s thesis, Arne Stormo, NTNU, 2009)

Work-flow in solving engineering problems

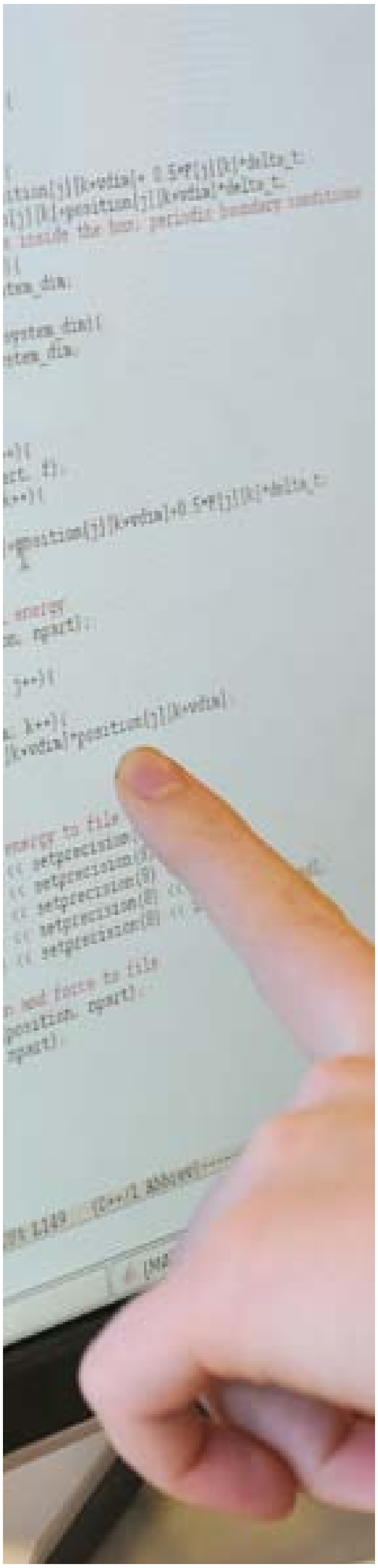
The usual work-flow in science and engineering disciplines is:

1. A problem is precisely defined in ordinary language.
2. It is then translated into a mathematically defined problem with the help of known laws and principles.
3. This is solved, very often with the help of computations, by computer.
4. he solution is visualised and interpreted in the context of the original problem.

All the steps require knowledge of the methods and assumptions which underpin the computations, and an insight of the possibilities and limitations of the methods and computations.

“How can a brain tumour be irradiated so that the tumour dies, but as much as possible of the surrounding tissue survives?”

[3] See Jakobsen et al. 1994



3.1.2 Renewal and co-operation

Universities and colleges are the most important institutions for developing competence. This type of competence is thus sourced by businesses primarily through staff recruitment. Teaching must be designed to safegurd this. Furthermore it is vital that business can keep up to date in a digital world that is constantly changing. New staff represent new knowledge coming into a company and can help ensure that renewal has sufficient quality. Computing will often underpin new solutions and working methods, and new employees with computational competence are therefore valuable assets.

Students should already be mature in computing competency when they enter employment. It is therefore natural to start with computing early in the studies so there is time for those skills to develop. An added advantage is that the students can be exposed to practical and realistic problems at an early stage. This in turn provides the students with sufficient competence to undertake their bachelor, master´s or doctoral degree project in a company. Computing oriented teaching can thus contribute towards increasing the co-operation between the teaching institute/ student and business.

3.2 Computing perspectives in education

- There is agreement that engineering and science subjects provide important foundations for today's society, and this is often used to motivate young people to study these subjects. Assertions about the usefulness of the subjects is often followed by a sentence "but this is too complicated to go into here", also to students who already have begun a science or engineering degree. Much of the cause for this situation is that students have no acquaintance with the tools and techniques employed professionally in the subjects, until near the end of their bachelor study, or in their master´s. An important reason is that these techniques are very often computing oriented, and in most degree courses this kind of topic is allocated to advanced subjects. The elementary subjects are mostly focused around classical paper-and-pencil methods, augmented with the use of calculators and similar tools.
- Education in science and engineering, as sketched out above, is well established in a solid, international tradition of education. The success of these subjects emphasises that this tradition has functioned well. The threshold for introducing major change should therefore be high, while at the same time central methods and solution techniques which are important in the execution of the subjects should be considered for inclusion in the foundation stage of the education. When this is done, emphasis must be placed on understanding of underlying principles, something which hopefully will become permanent core knowledge.

Computing applications in social science disciplines

Knowledge of computing has also become important in occupations which traditionally are not associated with science and engineering. Examples are marketing, research on social networks, environmental budgets. Specific problems can be:

- Should a contractor enter into a contract with roadstone firm A or B in connection with a major highway project, when environmental conditions are to be given top priority?
- Can one use information from a company's email system to identify and categorise the company culture with the help of quantitative analytical methods?

Education programmes in all disciplines thus face the challenge of how this perspective can be incorporated.

- Viewed from the central position computing has in business and research, it is high time that the foundation education in engineering and science at universities and colleges had a computational perspective. The basic principles the students need to learn are algorithmic thinking and problem solving, and not simply special software packages that build on these principles. Such software is nevertheless an important and necessary resource in the teaching environment.
- The following are some of the advantages of bringing computing into foundation level teaching in this way:
- Since the students gain early experience with the solution methods used in applying the subjects, the teaching can be made relevant from the start with respect to its professional use in a work situation. This will be motivating for the students and positive for their future employers.
- The road from the study of numerical methods to realistic applications becomes shorter, making it possible to improve the students' understanding and motivation for basic subjects like mathematics and statistics. This can be a key factor in shorter studies such as engineering.
- Computation competency makes it possible to study more realistic and advanced examples and problems early in the study. This can to advantage include examples where the complete theory is first explained later.
- The idea behind the numerical solution method is often simple and intuitive and can have a physical interpretation when applied to a physical problem. This is in contrast to analytical solution methods which to the students often appear to be a bag of tricks. Correct use of computing in teaching therefore has the potential to improve the students' grasp of the theory.

"How should people drive in heavy traffic to minimise queuing?"

Computing perspective in physics degree programmes

In physics degree programmes, integration of computations makes it possible to introduce the students early in their first degrees to realistic problems that lie closer to actual research. Mathematical problem solving is at the centre of modern physics.

Unfortunately only a few problems can be solved exactly, and without the help of more powerful computing methods the students must be limited to these. Most physics students therefore study classical problems such as a pendulum without air resistance, or a sphere which rolls down an incline. Even if these are well developed and iconic teaching examples in physics education, they appear far more idealised than realistic.

With integrated use of computing, the same physical principles can be studied with applications to more realistic problems, early in the degree programme. Instead of analysing the trajectory of a cannon ball without air resistance, the students themselves can work out a method for finding the wind speed inside a tornado by computing the path of a measuring instrument that is fired through it. This opens up working methods and ways of thinking which are central to the application of the subject both in research and industry. At the same time, the classic presentation of the discipline remains as important as before, and exact solutions of simplified problems are used where they provide good insight, but now they become part of a work-flow which better reflects what the students will encounter in graduate studies and in a future work situation.

4 Computing oriented education

"What will the sea level be in 30 years?"

4.1 What is computing oriented education?

Computing oriented education includes computations in a systematic and unified manner across the different subjects, so that the students gain a solid basis for using computing in their future careers. This implies, amongst other things, that:

- Computing fits in naturally, and can be co-ordinated, with both the underlying method subjects (mathematics and informatics) and the technical subjects in a degree course.
- The students develop an awareness of:
  - how computing and classic, mathematical solution methods together provide a powerful toolbox
  - of the limitations and various sources of error associated with the use of computing.
- The students develop practical skills both in the use of existing software and in programming calculations themselves.
- The students acquire a knowledge of the fundamental principles of how computers work. This is important for understanding both the limitations and possibilities of computing, and contributes to giving a general education which will be useful in later employment.

The idea is that the foundation in computing skills be laid as early as possible and be applied wherever it is natural. By laying the foundation early one will not need to spend time in later courses on teaching elementary skills.

4.2 What is traditional mathematics?

It is important to emphasise that the computing perspective must in no way be at the cost of teaching fundamental mathematics. On the contrary, an understanding of mathematics is more important than ever. Some examples of this are:

- Algebraic manipulation of expressions such that they can be effectively programmed.
- Familiarity with basic concepts such as the definitions of the derivative and the integral so that they can be programmed and calculated numerically.
- Transferring problems to a standard mathematical form so they can be solved with standard software.

4.3 Some aspects with introducing computing perspective

Major adjustments to the syllabus in the basic mathematics courses are rare, so very few institutions have routines in place for implementing changes on this scale. Below we give a short summary of some of the aspects which proved to be important during the implementation of a computing perspective at the University of Oslo. We also propose a national Centre which could support this work and make the transition easier at other universities:

- Early introduction
- Programming
- Choice of software
- Teaching materials
- Resources
- Adaptation to different subjects
- CSE Centre

"How can I know the effect of the advertisement I have on Google?"

4.3.1 Early introduction with practical training

The objective with introducing the computation perspective into higher education is that the students shall acquire knowledge about relevant computing methods, how these can be programmed, their characteristics and limitations, and their application in different fields. Therefore it is important that the computing perspective comes into the study as early as possible, preferably in the first semester. In that way computations may be utilised and extended in the subsequent semesters.

An overriding aim should be that the students themselves must program central methods. Experimenting with programs written by others can have its place, but often this induces a passive approach and is not very motivating. The methods should furthermore be related to the theory being taught in the first mathematics courses and used to solve realistic problems in applied subjects taken in parallel. The fact that fundamental methods are discussed in several courses will contribute to improved learning.

4.3.2 Programming

A computing perspective as sketched out above presupposes a fundamental knowledge about programming. In a world with crowded study plans it is easy to think that this will not take up much time and can be fitted in between other topics in regular subjects. The experience from UiO is that this does not work, and causes frustration for the students. It is therefore very important to allocate proper time for teaching the fundamental programming concepts in parallel with the first encounter with numerical calculations.

4.3.3 Choice of software

The core of a computing perspective is how problems can be solved with the help of precise algorithms. This is a fundamental concept which is independent of specific programming languages and computing platforms. At the same time it is clearly important that the students get practical experience with calculations on computers, something which does demand the use of a specific programming environment.

The choice of programming language often leads to heated discussion. It is therefore important to first agree on the overall demands one has for the programming language. This can vary, but some typical requirements are:

- The language must be suitable for focusing on fundamental programming.
- The language should use the fundamental number formats that are supported by standard hardware.
- A good plotting facility should be included.
- The language should not be too comprehensive, as that makes it difficult for the students to gain an overview of the language.

"What route should a district nurse take in order to fit in as many visits as possible in a working day?"

One consequence of the last point is that the language should not support symbolic calculations. The potential for symbolic calculations significantly raises the teaching threshold and makes it far harder to concentrate on the fundamental programming ideas. For example, it is easy to be confused by variables which sometimes are symbolic and other times numeric. While it is to the advantage of many students to be acquainted with symbolic computations, experience has shown that this is best done in advanced courses, when the fundamental programming skills have already been established.

4.3.4 Teaching materials

The textbook tradition in the sciences has developed over a long period, with good examples and exercises, and there is a wide choice of good textbooks. At the same time, there are very few textbooks with an integrated computional perspective as summarised above. Thus the most natural choice of teaching aids for a newly developed computing oriented course are a classic textbook supplemented with a compendium or book on numerical methods.

The disadvantage with such a solution is that the computing perspective simply becomes an add-on to the traditional course, both subjectwise and in the mindsets of the students. Looking a little further ahead, the aim must be to integrate the computing perspective with traditional theory. In that way the total volume will be reduced since computations will also be used to expand on the theory. At the same time, theory, modelling and computations will be perceived as an entity by the students. This means that introducing a computing perspective in many cases will involve developing teaching material.

4.3.5 Resources and costs

A computing perspective in education will be unfamiliar to most of the teachers and will lead to considerable extra work for everyone involved. Additional resources will clearly increase motivation, but are also a requirement for parts of the work. Once the computing perspective is well established, this kind of teaching will not necessarily be any more demanding on resources than traditional teaching, though that will depend on the teaching methods. At the moment it is difficult to say anything more precise about the cost of introducing a computing perspective – it will depend on local organisation, ambition level and much else. Some of the more important costs are identified below.

- **Remuneration.** The motivation for an extra effort can be increased if extra resources are made available in the form of increased travel funds, equipment budget or other form of compensation for the teachers. Similar incentives can also be given to institutes and other organisational units to speed up the work and encourage co-operation. Access to this kind of funding will ease the introduction significantly – the carrot principle always functions better than the stick!
- **Equipment.** A condition for succeeding is that the students and teachers have access to the necessary equipment. Today most students have their own laptop PC which they want to use when attending teaching sessions. In that case a routine will be needed for installing the necessary software on the students' machines. This is generally quite demanding since the diversity in the hardware and software can be great.
- **Raising compentency.** At most places there will be a large number of teachers with limited experience with computations on a computer. Getting these included within the reform process will be important both with regard to the collegial climate and in order to spread the workload. There will therefore be a requirement at many places for further training for the teaching staff.

4.3.6 Adjusting to different subjects

In most of this document, computing oriented education is described generically, without differentiating between different science and engineering subjects. The definition of computing used here nevertheless has a distinctly mathematical undertone, and it is therefore not surprising that computing oriented education will naturally take somewhat different forms for mathematical, and less mathematical, subjects. The general challenge is to provide a relevant foundation in mathematics and informatics, bound together by computations, and to then exploit this foundation in the secondary and tertiary subjects which come later in the degree programme.

"How should my home cinema equipment be set up to give the best possible sound reproduction?"

Mathematics and mathematical modelling are central tools in subjects such as statistics, the mathematical parts of informatics, the physics subjects (physics, astronomy and geophysics) and theoretical chemistry. These degrees therefore have a significant mathematical component, and the average student may be assumed to be interested in mathematics. This makes it relatively easy to introduce a computing perspective by including computational methods in the basic mathematics courses, typically numerical solution methods in addition to the analytical.

The professional practice of the mathematics oriented subjects at times requires using advanced methods from general computer science. For that reason it is an advantage if the training in programming at the start is similar to that given to informatics (IT) students, since some of the students in these subjects will want a significant informatics component in their study. At the University of Oslo this has led to the students in the mathematics oriented subjects being given a computing oriented programming course of 10 credits in the first semester, with the remainder of the semester devoted

"How should a finance institution place its capital to minimise risk but at the same time have good potential for a steady return?"

to classic calculus, numerical methods and digital representation of information; see more detailed descriptions in Section 5.1.

The less mathematical science subjects include physical geography, geology, biology, pharmacy , and parts of chemistry. Even if all of these subjects make use of mathematics, the extent is less, and the mathematical methods as a rule are less advanced than in the mathematical subjects.

In the traditional practice of these subjects, mathematical modelling is far less common than in the mathematics oriented subjects, but this is rapidly changing in step with the success modelling has had elsewhere. It is therefore important to give these students a relevant education in computing oriented modelling. The challenge from an educational perspective then is to teach computing to students who in general have little interest in mathematics and not very much mathematics from their schooling. A minimum solution is a course in elementary programming of 5 credits with a computing perspective, integrated with a 10 credit course containing relevant mathematics with a computing perspective.

Mathematics is often perceived as difficult and irrelevant by students in the less mathematics oriented subjects. An exciting possibility might be to let modelling, with the help of computing and programming, be the main focus, so that the mathematics is supplementing the computing, rather than the other way round.

4.3.7 Organisational conditions

A complete introduction of a computing perspective in education is so comprehensive that organisational and other human conditions are critical for its success.

The most important of such conditions are:

- Management support. As mentioned earlier, the introduction of a broad computing perspective demands co-ordinated changes to study plans within the field of foundational subjects such as mathematics, statistics and informatics as well as in various applied subjects. Such a process will necessarily involve many teachers with different backgrounds and competence. A prerequisite is therefore that the management of the section/faculty/institution first take the decision that these changes are needed, and then actively support their implementation.
- Steering group. During the establishment stage it would be natural to organise the introduction of computing through a project with a steering group which monitors the reorganisation closely. The objective must be that the computing perspective is included in the teaching on a permanent basis. The project organisation will therefore be temporary.
- Academic co-operation. An efficient introduction of computing demands co-operation across subject boundaries and between institutes and sections. In that way the applied subjects can exploit the foundation which is laid in mathematics and informatics without needing to spend time on introducing basic techniques. To achieve something like this it is important to establish a climate of co-operation across the traditional subject boundaries. This means that some personal hobby horses may have to yield to solutions of interest to a broader community, something which is rewarded in the form of a broad and modern programme which no single individual could implement on their own.
- Enthusiasm for the subject. An important basis for all good teaching is enthusiasm for the subject. A successful introduction of a computing perspective therefore depends on finding key staff with a passion for computing who are willing to

contribute beyond the normal call of duty. The work which will depend on these resourceful individuals will be to plan the subject content of central topics and the carrying out of the teaching in these subjects.

4.3.8 CSE Centre

Introducing a computing perspective into higher education is a major task in which there can be considerable benefit from internal co-operation within an institution, but also between the different institutions. The working group is of the opinion that it would be extremely advantageous if funding was allocated for establishing a national CSE Centre.

This centre would have the task of collecting relevant teaching resources and teaching experience in connection with introducing a computing perspective, initiate consequence research on the effects of the changes, and assist institutions during the changeover phase. The centre would also be responsible for arranging national conferences for sharing experiences.

5 Examples

"How can a milling machine be programmed to make a given groove in a sheet of wood?"

In this section we present three examples of how a computing perspective can be incorporated into foundation level degree programmes. The first example is from the Mathematics and Natural Sciences Faculty (MN) at the University of Oslo. This is the only example where computing perspective has already been incorporated and we can now report specifically about our experiences.

The two other examples describe how the incorporation is envisaged for NTNU in Trondheim and in the engineering degree programmes at the university colleges of Bergen, Narvik and Oslo. The reason why these institutions were chosen is simply that they are represented in the working group. Hopefully the ideas which are presented here will also be of interest for the country's other universities and colleges.

5.1 Mathematics oriented sciences at the University of Oslo

Calculations on computers have been a central tool within mathematics oriented sciences like physics, astrophysics, geophysics and chemistry at UiO ever since the first computers became available. For many years, however, this has been the preserve of research and was a tool that students for the most part had to learn on their own, usually in conjunction with the work for their master´s or doctoral thesis.

Around the year 2000 there were several groups, especially in the Physics, Informatics and Mathematics departments, which saw the need to introduce general training in computing already in the first semester of first degree courses. To begin with there was little synchronisation between these initiatives, but the establishing of the Centre of Mathematics for Application[4] in 2003 brought several of these communities together.

In 2003 there was also a major reform of higher education which had the effect that the first degree studies in mathematics oriented subjects were concentrated in a small number of broad bachelor programmes, the two largest being Physics, Astronomy and Meteorology (PAM) and Mathematics, Informatics and Technology (MIT)[5]. It was natural therefore that the previously scattered attempts to introduce a computing perspective were synchronised and systematised in the so-called CSE (Computers in Science Education) project. The description below summarises the project as a pilot scheme and reviews its status in 2011.

5.1.1 Establishment

The driving force behind the CSE project[6] has all along been academics who combine a strong commitment to teaching with computing oriented research profiles. Some of these figures were central in establishing the new bachelor programmes in 2003, and it was natural to attempt to establish a computing perspective for the mathematics oriented programmes. From the start our clear ambition was that CSE should establish an all-round computing perspective at first degree level, not limited to individual subjects and teaching programmes. There was considerable uncertainty at the time about how this could be done – no-one had any experience in implementing such broad changes.

It soon became clear that it was essential to have support for the project from the management of the MN Faculty. Key people in the project arranged a meeting with the Dean, with the result that work towards establishing a computing perspective in the degree programmes became an official item in the Faculty's strategic plan. With this backing, and combined with a respect for the classic content of the programmes, it was relatively easy to win the support of both the management in the various departments and the bulk of the teaching staff.

5.1.2 Structure

All the mathematics oriented programmes have a common block of courses in the first semesters where the computing perspective is established. This block consists of the five courses INF1100, MAT-INF1100, MAT1100, MAT1110 and MAT1120, which are described below. The foundation of this block consists of the three first courses which run in parallel in the first semester and form a natural entity:

- **MAT1100 – Calculus.** This is a classic mathematics course which includes topics such as complex numbers, convergence and continuity, differentiation and integration, as well as elementary vector calculus, functions of several variables and partial derivatives. In this course there is no systematic use of calculations on computers.

4. CMA, a centre of excellence funded by the Research Council of Norway.
5. Another major programme, Mathematics and Economics, was established from the 2007/2008 academic year.
6. For an introduction to computing orientated teaching, see the article "A nway to teach science? Computers in Science Education" on http://www.uio.no/for-ansatte/organisasjon/ikt_laring/fleksibel-laering/2009/

"How do you find a GPS position on the Earth?"

- **MAT-INF1100 – Modelling and computations.** This course contains three basic ingredients: Some classical mathematics; a number of numerical methods; and what may be called general digital skills:
  - Classical mathematics: induction, real numbers and completeness, difference equations, Taylor series with error term, ordinary differential equations.
  - Numerical methods: numerical simulation of difference equations, numerical solution of equations, numerical differentiation and integration, numerical solution of differential equations, derivation of error estimates, truncation errors and round-off errors in numerical methods.
  - Digital competence: Representation of numbers in different numeral systems, representation of numbers in the computer, round-off errors, representation of text, compression with special emphasis on loss-less methods, a brief introduction to representation and manipulation of digital sound and digital images.
- **INF1100 – Introduction to programming with scientific applications.** This course provides training in programming without any previous experience. The examples are mostly taken from mathematics and the natural sciences. Most of the numerical methods which are introduced in MAT-INF1100 are programmed in INF1100. The programming language is Python. The course starts with an introduction to elementary programming, but also gives a thorough introduction to object oriented programming.

The idea is that the students should experience these courses as a whole. A typical example is that the definition of the integral is discussed in MAT1100, numerical methods for calculating integrals are derived in MAT-INF1100, and these methods are programmed in INF1100. Comments from students indicate that this setup is demanding, but works well.

The other two courses in the foundation block are **MAT1110 – Calculus and linear algebra**, and **MAT1120 – Linear algebra**. These courses cover a fairly standard curriculum in multivariate analysis and linear algebra, but with a distinct computing perspective. Computing topics that are covered are fixed point iteration, solution of linear equation systems, numerical methods for finding eigenvalues and eigenvectors, the least squares method, singular value decomposition, as well as various applications such as image compression, Google's page-rank algorithm, and Markov chains.

Table 1 shows the structure of the Mathematical Finance specialisation in the bachelor programme Mathematics and Economics, while Table 2 shows the Astronomy specialisation in the bachelor programme Physics, Astronomy and Meteorology. Even though the specialisations are very different, both are built on the same foundation block. In this way all students acquire a common background in mathematics with a strong emphasis on computing perspective.

Some of the later courses also have a definite computing component. Within mathematical finance this is true for the two statistical courses, **STK1100 – Probability and statistical modelling** and **SKT1110 – Statistical methods and data analysis 1**, while all the courses in the astronomy specialisation include calculations to a greater or lesser extent.

5.1.2.1 Programming courses

Up until 2006 the training in programming was provided through a traditional informatics course, **INF1000 – Introduction to object oriented programming**, based on the programming language Java. In itself this is an excellent course, but the students clearly expressed that they wanted a course that was more closely related to the mathematics. A new course, **INF1100 – Introduction to programming with scientific applications** was introduced in the

6th semester	**EXPHIL03** Examen philosophicum	**Choice**	Choice of economics subject at 2000-level or higher
5th semester	**ECON3610** Resource allocation and economic policy	**MAT2700** Introduction to mathematical finance and investment theory	**MAT2410** Analysis II
4th semester	**MAT2400** Analysis I	**STK2130** Modelling by stochastic processes	**MAT2440** Differential equations and optimal control theory
3rd semester	**MAT1120** Linear algebra	**ECON2310** Macroeconomic analysis	**STK1110** Statistical methods and data analysis I
2nd semester	**MAT1110** Calculus and linear algebra	**ECON1500** Introduction to economics for natural science and mathematics students	**STK1100** Probability and statistical modelling
1st semester	**MAT1100** Calculus	**MAT-INF1100** Modelling and computations	**INF1100** Introduction to programming with scientific applications
	10 credits	**10 credits**	**10 studiepoeng**

Table 1. Programme option for Mathematical Finance in the bachelor programme Mathematics and Economics at UiO.

6th semester	**AST3210** Radiation I	**Choice**	**Choice**
5th semester	**FYS2160** Thermodynamics and statistical physics	**AST2120** The stars	**AST2210** Observational astronomy
4th semester	**FYS2140** Quantum physics	**Choice**	**EXPHIL03** Examen philosophicum
3rd semester	**FYS1120** Electromagnetism	**AST1100** Introduction to astrophysics / **GEF1100** The climate system	**MAT1120** Linear algebra
2nd semester	**FYS-MEK1110** Mechanics	**MEK1100** Vector calculus	**MAT1110** Calculus and linear algebra
1st semester	**MAT1100** Calculus	**MAT-INF1100** Modelling and computations	**INF1100** Introduction to programming with scientific applications
	10 credits	**10 credits**	**10 credits**

Table 2. Programme option for Astronomy in the bachelor programme Physics, Astronomy and Meteorology at UiO.

"How can the composition of a star be determined?"

autumn of 2007 in response to this. Feedback from the students indicates that they now experience the three courses in the first semester as a coherent whole.

5.1.2.2 Secondary subject – mechanics

So far we have described the foundation courses in mathematics and informatics. An important motive behind the teaching reform in Oslo is that the computing perspective introduced in these courses should be utilised in the various later courses which build on the foundation block. The relevant subjects are primarily physics, meteorology and oceanology, and astrophysics, in addition to mathematical subjects such as statistics and mechanics.

- FYS-MEK1110 – Mechanics is a central course with a distinct computing perspective. Thematically, it is a standard introductory course in Newtonian mechanics which encompasses topics such as kinematics, Newton's laws, mechanical energy, collisions, rotation and torque. The subject covers the classical themes and examples, but also seamlessly introduces numerical examples and methods, especially in connection with integration of the equations of motion. Thus one can do computations directly from experimental data and introduce more advanced force models. The programming language is Python or Matlab, and emphasis is placed on developing skills by considerable training in simple programming. The exercises vary from year to year, but an example is calculation of satellite orbits. In this exercise the students compute the orbits of known satellites and compare their results directly with measurement data from NASA.

"How can an aircraft land on autopilot?"

5.1.2.3 Modes of teaching

The introduction of a computing perspective in the first semester and the integration of computations in later courses has led to some changes in how the teaching is carried out. The greatest change is related to how exercises that contain programming are introduced and followed up. Such courses demand a certain infrastructure:

- A software platform must be developed which runs on the computer systems used in the teaching. This includes machines used in lectures, in computer labs, in terminal rooms for the students, as well as the students' and staff's own machines. At UiO we have chosen to support several platforms: a Python-based package for some courses, Matlab for others, and C++, Java or Fortran for more advanced courses.
- Premises are needed where students can work together in small groups on computing exercises, e.g. computer labs with PCs, or well equipped group rooms with electrical points and wifi, together with adequate AV equipment.
- Systems for handing in exercises that include text, mathematics, programs and visualisations. Standard computer-based communication systems have limitations in this respect.

Successful introduction of computing requires that the students develop both knowledge and skills. It is particularly important that computing skills are developed sufficiently, so that the students become able to work independently and can make efficient use of computing later in their education. This demands considerable practice over a long period. Computing courses therefore often include an element of exercises to be handed in.

In order to achieve a good learning outcome, computing exercises have been given in different ways: as a few larger project tasks or obligatory exercises with supplementary teaching in groups, and in a datalab following the pattern of science lab teaching; as numerous tasks integrated in the course and which demand group teachers and premises that are flexible for following up different methods; and as large projects which require suitable premises and group teachers. Experience indicates that it is easier to ensure that the students develop computing skills if this is also tested in the final examination.

Use of computing has made it possible to introduce more realistic examples and exercises in applied disciplines like physics, geosciences and astronomy. In many cases this requires that new teaching material is developed. A particular advantage is that computing gives the possibility of making exercises which more closely reflect the working methods used in research and industry, but this is dependent on having teachers with sufficient competence to set up and supervise such activities.

5.1.2.4 Examination results

It is important to stress that the basis for introducing a computing perspective is that calculations have become such a fundamental and central tool in the professional application of the subjects. In reality we therefore have no choice: our candidates must master this tool well. That said, clearly we must exploit the advantages of computing and include more realistic examples, and also try to provide more inspiring teaching. A natural question that arises is whether the students will then perform better in their studies and, more specifically, in their exams?

It is generally difficult to answer such a question for the simple reason that all students attending mathematics oriented studies at UiO now get a computing oriented education. In other words, there is no control group with which to compare. Nevertheless the following comments can be made:

- The study has become more demanding in that the students must relate to programming and a new method of mathematical solution, in addition to classical mathematics and science. In school, considerable emphasis is placed on paper-and-pencil calculation techniques applied to specific examples. Numerical methods may seem simple, but for students they are more abstract since they are general and typically are programmed for an arbitrary function *f*. These methods can therefore not be learnt with the help of calculation techniques in the way the students have been used to doing. The fact that numerical solutions are the "usual" solution type for problems appears to be a major hurdle for the students – it is only in special cases that can one find a solution given by an explicit formula.
- Examination results and the failure rate in the foundation block courses is about the same as before the computing perspective was introduced. There may be many reasons for this, but altogether it seems like the students master the computational orientation quite well.

5.1.3 Computing in other subjects

The introduction of a computing perspective in the basic education at the MN Faculty at UiO was initially restricted to mathematically oriented subjects. This was because these subjects had a group of teachers who were aware of the need and took the initiative to reorganise the courses. However, the Faculty management have long had the aim to introduce the computing perspective into all the science subjects, and this was written into the Faculty's strategic plan for 2004-2009. It must be emphasised that there have been individual course in other subjects with computing components, and many study plans have included introductory courses in programming, but this as not been coherently integrated into the basic education.

The Department of Geosciences includes mathematically oriented disciplines like meteorology and oceanography which have been involved in the CSE project from the

start. Initially, this was restricted to the foundation courses in the bachelor programme, Physics, Astronomy and Meteorology, which were given a computing perspective already in 2003. An integrated computing perspective was then introduced into the more advanced courses from 2008.

The Autumn 2008 saw the launch of **GEO1040 – Introduction to programming with applications in geosciences**. This course provides an introduction to elementary programming based on examples from geoscience. The course does not go into the more advanced topics in INF1100 (the programming course for the mathematically oriented students) and is not co-ordinated with a mathematics course. GEO1040 has been met with great interest from disciplines such as biology and chemistry who would like to offer a comparable course. The basic challenge is to develop a relevant and synchronised foundation in mathematics and informatics for all the less mathematically intensive disciplines.

"How can heart specialists become skilled surgeons without practicing on people and animals?"

5.1.4 An important bonus: increased interest in teaching

The work to establish a computing perspective in the education offered at UiO has been time consuming for the people involved, with much discussion across traditional scientific boundaries. While some of the discussions have been related to mathematics and science, some has been more focused on the teaching challenges. In addition the CSE project has organised an annual CSE seminar which in recent years has attracted about 50 participants. The main purpose has been to discuss computing perspective in our education, but the seminar has evolved to also include more general teaching themes.

At most universities it is generally difficult to get scientists seriously interested in pedagogic questions. However, introduction of a computing perspective is primarily a mathematical and scientific challenge which seems to interest a significant part of the staff. Reflection around teaching reforms, evaluation modes and such like are natural consequences of this work, and for many lecturers this has resulted in greater interest in teaching in general.

5.2 NTNU

5.2.1 Historical background

Since the founding of the Norwegian Institute of Technology (NTH), Trondheim has had a national responsibility for engineering subjects. This tradition has been continued by the Norwegian Technical and Scientific University (NTNU). NTNU was formed in 1996 by the amalgamation, amongst others, of NTH and the Allmennvitenskapelige Høgskolen (AVH); the latter had been organised according to the classical university model.

The result is that today NTNU offers 5-year integrated master´s degrees in technology/civil engineering as well as bachelor and master´s degrees in the natural sciences. In the mainstream natural science subjects – physics, chemistry and mathematics – NTNU offers both degrees. The students choose a study programme with associated degree when they apply for admission.

5.2.2 The study's structure

The main content in the technological studies at NTNU is specified by the *Management Committee for Civil Engineering Education* – FUS committee[7]. This committee also influences the natural science studies indirectly, since the majority of the science programmes are built upon the same courses as those offered in the technology programmes.

All the technology studies at NTNU are built up around the same template: one part consists of technical and non-technical subjects which are common to all the study programmes, and another part consists of specialisation subjects. As an example, virtually all 1500 technology students will attend the two courses Mathematics 1 and IT Foundation in the first semester. The following semester the bulk of the students attend Statistics and Mathematics 2-4. In addition, all the technology programmes typically include at least one foundation course in physics.

5.2.3 Today's situation

International evaluations of both civil engineering studies in general and specific study programmes in particular have in recent years called for more focus on computing. Norwegian industry has done the same, since today's engineers increasingly use numerical simulations in planning, evaluation and testing of technical designs[8].

As a result of this feedback several working groups have been set up, both centrally and at individual faculties/departments, in order to plan how one can best introduce more use of numerical computing in the technical and natural science educations. So far this has unfortunately not resulted in more than a handful of practical changes. Nevertheless, a number of teachers at NTNU who see the importance of computing and feel passionate about the subject, have already introduced this in their teaching. The risk is that it will be dropped again if there is a change of teacher.

5.2.4 How to introduce CSE?

The working group's clear opinion, and recommendation, is that when a computing perspective is to be introduced into an education programme, it is vital to ensure predictability and continuity both with regards to what the students learn, in which courses the learning takes place, and the order in which the material is learnt. Without these elements in place, the students' assumed knowledge at any given level will be diffuse and uncertain. In particular, the knowledge will be liable to vary from year to year, which would make it challenging to plan and carry out numerical computing in a given course within the strict timeframes that apply.

Since computing is such an important part of modern engineering and natural science, we think it should be introduced at NTNU in all the more than 25 different study programmes in these disciplines. The introduction should happen in both the general courses and the specialisation courses. In the general courses, which typically come in the early part of the study, one should focus on general computing and programming skills. The specialisation courses, on the other hand, provide an opportunity for taking a more subject-specific approach to problem solving and the computing tools (software packages) needed for this.

When introducing computing into a discipline, one often comes up against the problem that not all the scientific staff are equally familiar with the new material. This can make it difficult to maintain continuity in a structured plan for computing, since it will be vulnerable to changes in teaching staff or other smaller changes. Our opinion is therefore that it is more practical and appropriate that the main responsibility for computing does not lie primarily with the lecturer of a course. It is obviously desirable, and expected, that lecturers contribute scientifically related problems which lend themselves for computing activity in their course. However, the practical planning and implementation of this activity should not be the lecturer's responsibility.

7. See http://www.ntnu.no/adm/utvalg/fus (in Norwegian)
8. See Computational Science and Engineering: Challenges and Opportunities Contributions Towards a Centre for Computational Science and Engineering at NTNU and SINTEF, NTNU 2010.

"What will be the effect if part of a mountain collapses into a fjord?"

5.2.5 General courses

The two courses, Mathematics 1 and the IT Foundation course, are compulsory for virtually all technology students at NTNU in the first semester, and we specifically recommend that a computing component is introduced in both courses. This means that the courses must be co-ordinated. Both should include computing tasks that are based on problems, or require using techniques, from the other course. Mathematics 1 ought to have tasks that require the use of a computer to carry out the calculations, while in the IT foundation course there should be tasks which are motivated from the mathematical world. The subsequent common courses in mathematics (Mathematics 2-4), statistics and physics should integrate computing in a similar way, and later courses in programming also ought to have tasks linked to mathematics, natural science and/or engineering disciplines.

The word "tasks" is used above in the wider sense of an activity which is a part of the course. Examples may be problems which the students must solve, and maybe hand in, and small projects or semester projects.

The specific form chosen in a particular course will depend on the character and nature of the course. For early courses, it will probably be preferable to integrate computing in the compulsory weekly problems. For advanced courses other forms can be used to advantage. Irrespective of which mode is chosen, serious consideration should be given to making the computing activities compulsory. This ensures that all the students solve them in a satisfactory manner and with acceptable results.

5.2.6 Specialisation courses

Computing should also be integrated in the different specialisation courses. It is recommended that most (if not all) specialisation courses should have at least one computing oriented task on a theme which is relevant to the course. This means that only a minor adjustment of each individual course is needed in order to introduce computing in the various study programmes. This should make it possible to introduce computing relatively quickly in the specialisation courses.

5.2.7 The main idea behind the introduction of computing

The main idea behind the plan sketched and recommended above is that the total sum of computing and related methods learnt in the individual courses should constitute a planned entity. When the study is completed, the accumulated computing activities will have given the prospective NTNU engineers and natural scientists an experience base and a toolkit enabling them to satisfactorily carry out, understand and interpret the results of numerical. The hope is that computing will simply have become a natural part of solving a problem!

To achieve this the study programmes need to identify relevant types of computing and the extent of these for their students. Further, based on the different character of each individual course, and the desired sequence of the computing methods, one needs to plan which computing topic should be discussed in which course. The most elementary computing topics should be covered in the general courses such as Mathematics 1-4 and the IT courses.

5.2.8 Computing laboratories

To ensure continuity, one should seriously consider whether a group of permanent teaching staff should be given the overall responsibility for the computing in the different courses. Assuming they have the necessary competence they can take on such a role as part of their regular teaching. In this way one is less vulnerable to varying levels of knowledge about computing amongst the lecturers who happen to teach a given course.

In several departments one already has such an arrangement for the compulsory experimental laboratory work which forms part of some courses. Here we are thinking in terms of a similar organisation of the computing part, in other words we recommend the introduction of "computing laboratories".

5.2.9 What about the science students?

As mentioned earlier, there are some natural science fields at NTNU (e.g. physics and chemistry) where both technical and natural science study programmes are offered, and where many courses are taught collectively for both the programmes. It is important that also the natural science students acquire the necessary programming competence demanded for carrying out the computing in the specialisation courses. The introduction of computing must also lead to necessary adjustments of the demands and recommendations for course combinations in the traditionally more free natural science programmes.

5.2.10 Challenges

Each autumn more than 1500 new NTNU students attend the Mathematics 1 and IT Foundation courses. Because of this large number, guidelines and specific plans need to be formulated for how to deal with many of the practical problems which will arise. Specific problems are:

- Should the students be required to have their own computers? If so, what about those who do not have one?
- How does one ensure that working software is installed on the machines of all students and also in the computer rooms?
- Are there enough electrical points and network connection points in the rooms where student supervision is to take place?

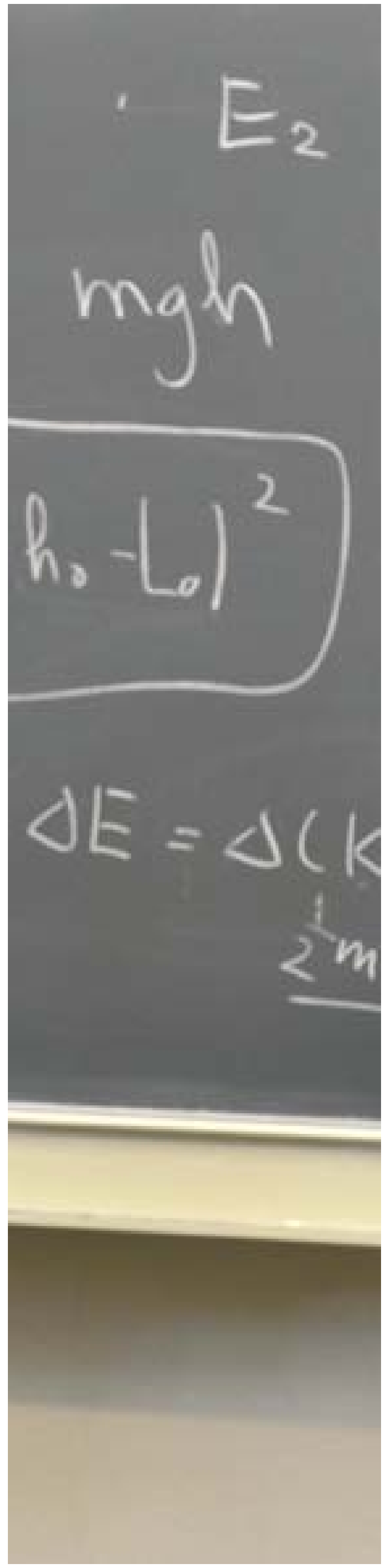
The solutions to such practical problems often have unforeseen effects on how one can develop and carry out a good and functional plan for use of computing.

As mentioned earlier, one challenge is that some of the science staff may not have enough knowledge about computing. There may therefore be a need to set up special courses in computing for scientific employees.

5.2.11 Summary of recommendations for NTNU

The core recommendations made above for introducing computing at NTNU may briefly be summed up in the following points:

- General courses, i.e. Mathematics 1 and the IT Foundation course, are used to introduce a common co-ordinated platform for computing.



- All specialisation courses throughout the study, where relevant, should contain a computing component which consists of course-relevant and compulsory tasks with the objective of teaching the students specific computing techniques.
- Computing laboratories should be set up for each study programme with appointed "laboratory responsible" person, consisting of members of the scientific staff who have computing competence.
- The total computing activity in a study programme should provide a well-founded entity. Changes shall only be made exceptionally and only after thorough evaluation (since this can often have unforeseen consequences for later courses).

5.3 Engineering education in Oslo, Bergen and Narvik

5.3.1 Background and general considerations

"How is it possible to obtain three-dimensional images of the inside of the body?"

Engineering education in Norway is at the threshold of far reaching changes in the form of new regulations and a new framework plan which will come into effect at the beginning of the 2012-2013 academic year. This means that the Colleges must go through all their study programmes/specialisations and courses and amend these in line with the regulations. The time is therefore ideal for also including and integrating a computing perspective in the engineering studies, since the plans have to be revised and adapted in any case.

Some study programmes are less mathematics and computing oriented, and some people will claim that it is a waste of time teaching programming and computational mathematics to these students. Here the same argument applies as in all the other subjects and topics in engineering education. Norwegian engineers today should have sufficient competence within classic engineering subjects to be able to communicate with other engineers about problems which they themselves do not work with day to day.

We maintain in this report that computational mathematics has become a classic engineering subject, and that it therefore is necessary for all engineers to be competent in this field. In addition the use of these methods is increasing amongst practitioners in other disciplines that traditionally have not been computationally oriented.

5.3.2 Integration of computational mathematics in the study plans

We recommend that students complete a course in programming as early as possible in their studies. In this way a foundation will be laid for increasing computing competence through the study, preferably with a start being made early in the first semester. According to the new framework plan there is scope for this both in what is called general courses and in the more specialised programme courses.

General courses constitute a group that comprise 30 credits and will be common to all the study programmes. In the national guidelines for engineering education it is stated that, "This group of courses has the purpose of contributing motivation, completion, overall understanding, and system thinking as well as a mathematical foundation for the education".

The engineering oriented introductory course (10 credits) is placed in the first semester, together with Mathematics 1 (10 credits) and a specific programme course (10 credits). For the latter course the national guidelines give the following description and suggestion for relevant topics (quote):

Project work, report writing, presentation technique, bid and contract writing, ethics, health, environment and safety, lifespan analyses, project budgets, laboratory work, computing perspective with help of computers, use of algorithms and mathematical computing with computers.

The programme courses are common for students within a specific study programme. This course group has 50 credits and consists of the core subjects in the study. The core subjects should provide a fundamental introduction to central programme topics, and enable the candidates to specialise within their programme area.

Our understanding is that it would be best if the programming skills were taught as a general course, and that the programme courses were specific for each study programme. This can however be adjusted. What is important is that the students gain programming skills in the first semester and that this is further developed in the courses specific to the different study programmes throughout the study.

It is important that the computing perspective is integrated in all the relevant disciplines and courses, both in programme courses and in the more specialised courses. To achieve this the lecturer must set compulsory work that includes a computational perspective both with respect to the use of solution tools and methods.

5.3.3 Educating the staff

If lecturers have little experience with numerical computations and computational mathematics, the institutions should arrange for assistance and tuition. For example, this can be solved by the institution setting up a computing laboratory where a laboratory responsible person is employed who will provide support for the lecturers with insufficient competence within computational mathematics. Lecturers and the laboratory responsible person can together find one or more tasks within the course that are relevant for teaching and learning.

It is not desirable or intended that the mathematics or other curricula should be reduced or changed significantly beyond the requirements of the new regulations and framework plan, in order to make room for computational mathematics. But there will be topics and courses within the individual disciplines which will be better suited than others for using computers.

5.3.4 Teaching benefit

It is important for the students' knowledge, skills and general competence that using a computer as a problem-solving tool becomes an integrated part of all teaching at the bachelor level. This will be a benefit to engineering students when they start their main project and if they proceed to the master´s level.

The master´s programmes may have more time available since the students will already know much of the computing that is taught at the master´s level today. We believe this to be an advantage for the engineering candidate and a bonus for the firms that will employ the engineers in the future.

"How can we make a new material with a specific property?"

5.3.5 Strategically led by the teaching management

We believe that the faculty and department managements and those with overall responsibility for courses and teaching must play an active role in integrating and introducing computational mathematics in the individual study programmes and courses. This is a quality assurance that the integration is carried through even if lecturers leave or change subjects

5.3.6 Costs

We assume that the introduction and integration of computing mathematics will lead to increased use of computing equipment, both that owned by the college and that owned by the students. This can lead to increased costs in the form of increased need of hardware, more time spent on technical assistance with installing software, more power sockets for laptops in classrooms, etc. It is possible to minimise this with carefully thought-out procedures and written instructions made available for everyone. It is therefore important that the management at the college and section levels are aware of this, so it gets included in their strategies, action plans and budgets. At the same time we view it as an advantage that introduction and integration of computational mathematics occurs simultaneously with the introduction of the new regulations and framework plan for engineering education in Norway. This means that the extra costs associated with planning and changes in the study programmes and study plans are minimalised in comparison to an introduction of computational mathematics into today's bachelor degrees.

5.3.7 Recommendations for engineering education

- The teaching management at each college must take responsibility and be actively involved in the discussion of how computational mathematics should be adapted to the teaching at the individual institution.
- The costs associated with adapting to more computationally oriented mathematics in the bachelor programmes must be included in the budget process.
- The colleges should ensure that they integrate computational mathematics simultaneously with adapting the teaching to the new regulations and framework plan for engineering education at bachelor level.
- An introductory course in programming should be part of the first semester, and co-ordinated with Mathematics 1. The institutions should set up a computing oriented laboratory to secure competence and continuity in the teaching where computational mathematics is a part. The engineering programmes should plan to include a computing perspective into all courses of study. The computing perspective should be introduced into all the relevant courses.

6 Closing comments

The working group was given a mandate by the Ministry of Education and Research (KD) to produce a "model for how the CSE Project (Computers in Science Education) can be brought into other education establishments within MST disciplines", and more specifically to "compile a written guide and contribute to information dissemination through activities which support the named objectives [about increased quality in Norwegian MST disciplines]". The mandate names three specific points which should be discussed in the guide.

We believe we have answered why and how a computing perspective should be incorporated within a broad spectrum of teaching programmes, and how this task can be organised. Many lecturers will find useful advice about implementing a computing perspective when they read this guide.

Certain of the more demanding requests in the mandate have not been answered in the same detail. We found the most complex question concerns resources and costs. It is difficult to separate work time and other resources used for introducing a computing perspective from the general teaching effort and maintenance of the teaching programmes.

As we have said earlier, major revisions in science disciplines happen perhaps a couple of times in a century. Now that we find ourselves taking part in such a revision, it can be seen as both a special responsibility and a unique opportunity to participate in forming society's future.

7 References

Computational Science and Engineering: Challenges and Opportunities Contributions Towards a Centre for Computational Science and Engineering at NTNU and SINTEF, NTNU 2010

Hjort-Jensen, Morten, Knut Mørken, Annik Myhre and Hanne Sølna (2009): ”A new way to teach science? Computers in Science Education”, page 29-40 in Susanne Kjekshus Koch (red.): Ringer i vann. Lenge leve fleksibel læring ved Universitetet i Oslo, Universitetets senter for informasjonsteknologi, Universitetet i Oslo

B. Jakobsen, B. and F. Rosendahl: “The Sleipner Platform Accident” in *Structural Engineering International 4(3)*, 1994

Norwegian Ministry of Education and Research (Kunnskapsdepartementet) (2010): Realfag for framtida. Strategi for styrking av realfagene 2010-2014

Ny forskrift om rammeplan for ingeniørutdanning: http://www.regjeringen.no/nb/dep/kd/dok/andre/brev/utvalgte_brev/2011/ny-forskrift-om-rammeplan-for-ingeniorut.html?id=632491

Stormo, Arne (2009): Integrering av numeriske beregninger i grunnleggende fysikkurs, Masteroppgave i teknisk fysikk, Norges teknisk-naturvitenskapelige universitet, Trondheim

8 Photo acknowledgements

Francesco Saggio:
Page: 1, 5, 12, 20, 25, 27, 29, 31, 36

Colourbox.com:
Page: 10, 13, 14, 19, 22, 33

ScandinavianStockPhoto
Page: 8


Applications Places System
Computer
Terminal
sea.png
Trash
Prosjekt2.tex
Prosjekt2.pdf
Prosjekt2.log
File Edit View Project Build
pcc.cpp
argon.cpp