CSE for Engineers

Patrik Jansson, Dept. of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

November 21, 2009

1 Introduction

The Computer Science and Engineering (CSE) department gave me the task to

- survey the needs for CSE-education components in the Chalmers masters programmes where CSE is not a core of the curriculum;
- based on the survey suggest suitable courses or parts of courses to fill those needs; and
- (possibly) suggest changes in the current CSE curriculum to make it easier to enter into the CSE field from outside

Due to my involvement in the Chalmers eScience Initiative I have chosen to broaden the scope somewhat to also include *Computational* Science and Engineering and courses outside the CSE department.

2 Definitions and terminology

For this document I define the "core CSE" to include the following programmes:

- "IT" (Software Engineering) and "D" (Computer Science and Engineering), 5-year programmes
- All the seven CSE MSc programmes:
  ID = Interaction Design (MPIDE),
  SET = Software Engineering and Technology (MPSEN),
  CSALL = Computer Science: Algorithms, Languages and Logic (MPALG),
  SDCS = Secure and Dependable Computer Systems (MPDCS),
  NDS = Networks and Distributed Systems (MPNET),
  IESD = Integrated Electronic System Design (MPIES),
  CS = Computer Science (N1COS at GU)
- Two AIT MSc programmes:
  ISD = Intelligent Systems Design (ITIDM)
  SEM = Software Engineering and Management (N2SEM at GU)

Figure 1: Student opinions word cloud

3 Student opinions

I started with going through the Chalmers-wide masters student questionnaire from May 2009 in search for free text comments about courses the students think should be added. Below I have listed and visualised (in Fig. 1) the most common words in the CSE-related answers.

20*design; 16*programming; 15*network; 14*control; 12*system; 10*Network; 10*advanced; 10*software; 7*theory; 6*basic; 6*computer; 6*Flash; 6*practical; 6*security; 5*Finite; 5*prototyping; 5*related; 4*almost; 4*applied; 4*approach; 4*architecture; 4*classes; 4*elective; 4*element; 4*embedded; 4*engineering; 4*FEM; 4*information; 4*knowledge; 4*machine; 4*systems; 4*maybe; 4*PLC;

Based on this material I searched the Chalmers Student Portal[1] and identified ten topics (many directly corresponding to courses) which could be relevant for further investigation.

4 Opinions of Directors of Masters Programmes

The topics were used in a poll sent to all the programme directors asking them the "vote" for subjects important to some of the students on their programmes. I got 16 proper votes + a few emails, but here I only count the votes from the 12 directors outside "core CSE". The coverage is only 12 out of around 40, but this may also be indicative of how (ir-)relevant the CSE subject was to the other 28 directors. The resulting topics list in order of decreasing votes is as follows (see Fig. 2 for the details): 10*Matlab, 9*Scientific vis., 7*C/C++, 4*Java, 4*Databases, 4*High perf. comp. (HPC), 3*Alg. and datastructures, 2*Par./multicore prog., and no votes for Data com. nor for Internet tech.

I also received free-text comments about courses missing from the list: two mentioned "Human-computer interaction" for masters students; one mentioned Software engineering, one Computer graphics and one explained that "We're interested in generic modeling capabilities but not language-specific knowledge, i.e. not an advanced Matlab course".

[1] `http://www.student.chalmers.se/`

Enkät "CSE for Engineers"

http://doodle.com/wqcg66x4ees76yy8

Dept.	Masters prg. & person	#	Prog./model (matlab)	Scientific vis.	Prog. (C/C++)	Prog (Java)	Databases	High perf. Comp.	Alg.&Datastr.	Par./multicore prog.	Data com.	Internet tech.
FF	MPPHY: Christian Forssén	6	1	1	1			1	1	1		
S2	MPSYS: egardt	6	1	1	1	1	1		1			
PPU	MPPDE Andreas Dagman	5	1	1	1	1	1					
MC2	MPNAN Göran Johansson	5	1	1	1			1		1		
S2	MPCOM Thomas Eriksson	4	1	1	1	1						
TF	MPCAS	4	1	1	1			1				
KB	MPMNT: gorans	3	1	1		1						
KB	MPCBS Jerker	3	1	1				1				
EoM	MPSES David Pallarès	3	1				1		1			
A	TKARK Michael Edén	2		1			1					
MC2	MPMPE, Piotr Starski	2	1		1							
BoM	MPSTR Mario Plos	0										
	Antal	43	10	9	7	4	4	4	3	2	0	0
ITIT	MPISD, Claes Strannegård	3			1	1			1			
ITIT	N2SEM Miroslaw Staron	5	1		1	1	1		1		1	
	Antal	51	11	9	9	6	5	4	5	2	1	0
DoIT	MPNET Tomas Olovsson	4			1	1			1		1	
DoIT	MPALG Peter Dybjer	3				1	1		1			
	Antal	58	11	9	10	8	6	4	7	2	2	0

Comments

Olof Torgersson	Missing: "Human-computer interaction" for masters students
Johan Malmqvist	Missing: "Software eng.", "Comp. graphics", "Human-Comp. Interaction". Important: "Java", "Databases", "Internet tech." (rather Internet prog.)
Mario Plos	It might be of interest occationally to sngle students, but not every year
Christian Forssén	We're interested in generic modeling capabilities but not language-specific knowledge, i.e. not an advanced matlab course.

Figure 2: Programme Directors' votes

5 Existing courses

Based on the priorities of the directors I've looked at what the CSE area has to offer currently. I've tried to identify all matching courses but I may have missed a course or two. Each course is described using its learning outcomes according to the student portal (quoted verbatim without making the spelling match the rest of the document).

5.1 Programming / modelling (Matlab)

Math offers TMA325 - Introduction to engineering mathematics:
Learning outcome (after the course, the student should be able to)

- operate in the environment at the department of mathematics
- use mathematical software and texteditors such as MATLAB, emacs, and LaTeX.
- profit by the education in the courses to follow and the masters's thesis

This course is formally compulsory for starting students on the Engineering Mathematics MSc but according to the programme director it is only for the external students (those who did not do a BSc at Chalmers). There are a few other Matlab courses on the BSc level, and a few fragments inside other MSc courses, but there is no other full course in English.

5.2 Scientific visualisation

Math offers MVE080 - Scientific visualization:
Learning outcome (after the course, the student should be able to)

- think in visualization terms
- produce insightful graphics in a number of common and important cases
- use advanced Matlab graphics
- construct graphical user interfaces in Matlab
- use OpenGL and OpenDX to some extent

This course is an elective (advanced) MSc course offered primarily to Eng.Math. students in their second year.

5.3 Programming in C/C++

There seems to be no MSc level programming course in C / C++ offered at all.

The GU Physics department is offering related BSc level courses in Swedish FYD420 - C++-programmering I, 7.5 hp and FYD430 - C++-programmering II, 7.5 hp. Descriptions (in Swedish):

C++-prog. I Kursen har som mål att ge dig kunskaper i programmering av moderna mätsystem samt uppbyggnad av dessa. Du utnyttjar det objektorienterade programspråket C++ för att utveckla Windowsbaserade mätapplikationer samt analysprogram för detta och inhämtar kunskaper i bl a dynamiska länkningsbibliotek. Kursen ingår i utbildningsprogrammet Datorstödd Fysikalisk Mätteknik.

C++-prog. II Kursen är en direkt fortsättning på C++-programmering 1. Du utnyttjar LabWindows/CVI och Visualstudio C++ för att utveckla avancerade mät- och styrprogram. Kursen ingår i utbildningsprogrammet Datorstödd Fysikalisk Mätteknik.

5.4 Programming in Java

CSE is offering DAT160 - Programming for science:
Learning outcome (after the course, the student should be able to)

- have knowledge of concepts of computer programming, the Java language and class libraries, object-oriented programming and structured program design.
- be able to write Java programs consisting of classes made by the student, as well as standard classes for graphical user interfaces and data structures.

This course in an compulsory for students with non-CS background in the Bioinformatics and systems biology MSc.

5.5 Databases

CSE is offering TDA357 - Databases (at the BSc level):
Learning outcome

- By the end of the course the student should know how to design and use databases as an end user, as an application programmer and as a database administrator.

This course is listed as recommended elective in several MSc programmes: Bioinformatics and systems biology; Biomedical engineering; Industrial engineering and management; Interaction design; Systems, control and mechatronics.

5.6 High performance computing (HPC)

Math offers TMA881 - High performance computing:
Learning outcome (after the course, the student should be able to)

- describe the basic features of a modern CPU and analyze how these affect the performance of a code
- optimize code written in Fortran or C
- optimize Matlab code
- use basic unix tools such as make, gprof and ld
- mix Fortran, C and Matlab (mixed language programming)
- write simple parallel programs using MPI and OpenMP

This course is an advanced elective in the Engineering Mathematics MSc. It has the following prerequisites: "Basic courses in mathematics, numerical analysis, programming and data structures. Basic Matlab programming."

5.7 Algorithms and datastructures

CSE is offering TIN092 - Algorithms:
Learning outcome (after the course you will be able to)

- Model problems: Formulate a clear mathematical model of real world problems.
- Design algorithms: Construct algorithms with several algorithm design methods and be able to choose appropriate data structures and abstractions.
- Describe your algorithms and their qualities to others.
- Prove correctness: Prove that your algorithms are correct.
- Analyse algorithms: perform an objective evaluation of the performance and be able to compare it to other algorithms performance.
- Recognize intractable problems and other classes of problems like P, NP, NPC
- Recognize a number of specific algorithms for basic problems which tend to appear over and over again in different applications

. . . [ed. A few sentences cut out]

This course is compulsory in the CSALL and SET MSc programmes but also recommended elective in IESD, NDS, and Systems, control and mechatronics MSc. There are other related BSc level courses as well, but they are in Swedish.

5.8 Parallel / multicore programming

CSE is offering TDA381 - Concurrent programming (BSc level):
Learning outcome (after the course, the student should be able to)

- apply practical knowledge of the programming constructs and techniques offered by modern concurrent programming languages. This includes the ability to identify synchronization problems, design and argue for the correctness, clarity, and efficiency of solutions, as well as implement such solutions in expressive programming languages.

The student should also be able to demonstrate the critical knowledge of:

- shared-memory and message-passing models,
- problems that arise in concurrent systems (such as shared update, dining philosophers, producer-consumer, resource allocation, time-outs), and
- common patterns for solving them (such as locks, client-server, pipelines, replicated workers, barrier synchronization, passing the baton, alarms)

This course is recommended elective in NDS and SET.

5.9 Human-computer interaction

CSE offers DA288 - Human - computer interaction (BSc level):
Learning outcome

- After this course, the students shall be able to solve certain interface design problems and also to go deeper into any specific design area by taking advanced courses or even by self-study.

This course is compulsory for BSc Software Engineering ("IT") students and elective for the Biomedical engineering and Systems, control and mechatronics MSc programmes.

6 Entry points into the CSE core education

For *BSc students* in non-CSE subjects, the recommended path is what the BSc programme "I" (Industrial engineering and management) defines as their "Information technology track". (This "IT-track" is specially designed for entering the SET MSc programme and the last two courses could perhaps be modified to better suit other CSE MSc programmes.) The track assumes a basic programming course in Java (comparable to "Programming for Science" in section 5.4) as a prerequisite and contains the following courses:

- TDA550 - Object-oriented programming, advanced course (BSc level, in Swedish)
- LET375 - Algorithms and data structures (BSc level, in Swedish),
- TDA357 - Databases (see 5.5, BSc level, in English),
- EDA496 - Software Engineering Fundamentals (BSc level, in Swedish)

For *MSc students*, on the other hand, this path does not work in its current form because three out of one+four courses are taught in Swedish. A possible replacement course to use instead TDA550 is TDA451 - Functional programming. So perhaps it could be enough to develop one "Datastructures" course and one (basic) Software Engineering course in English.

7 Summary

MSc programmes outside the CSE core identified the following needs:

- programming / modelling in Matlab, C/C++ and Java,
- Scientific visualisation
- Databases
- Algorithms, datastructures and parallel / multicore prog.

The needs are mostly fulfilled with existing courses (only C/C++ missing), but it seems that too few teachers and students know about them. On the other hand, if a broader range of students start taking these courses the concrete contents may need some negotiation. There is an inherent tension in courses being offered both to the BSc and to the MSc level.

To enter the CSE area from a non-CSE background there is a need for a MSc-level continuation after "Programming for Science".

Some more material is available on the CSE wiki page on CSE for Engineers.