

eValuering 2.1 - Sammanställning

SAMMANSTÄLLNING

Utvärdering: Constraint Technology (autumn 2008)
Antal svarande: 18





WELCOME!

Please fill out the survey below to provide the department with feedback. Don't forget to click on "Submit Evaluation" when the form is complete. Your answers are anonymous.

QUESTIONS

The following questions are required by the faculty board. You must answer them.






What is your general feeling about the course?

Svarsalternativ	Graf (%)	%	Σ
1 (= bad)		0	0
2		6	1
3		11	2
4		39	7
5 (= good)		44	8



Medelvärde: 4.22

Standardavvikelse: 0.88



The total amount of work on the course, in relation to the credits (7.5 högskolepoäng = 7.5 ECTS credits = 200 hours of work expected), was ...

Svarsalternativ	Graf (%)	%	Σ
1 (= low)		11	2
2		6	1
3		44	8
4		33	6
5 (= high)		6	1
Medelvärde:	3.17		
Standardavvikelse:	1.04		

Did you at the start of the course receive information about previous course evaluations and measures taken because of them?

Svarsalternativ	Graf (%)	%	Σ
no		17	3
yes		83	15

Did you get the opportunity during the course at a scheduled time to give anonymous written feedback on the ongoing course (in short: was there a mid-course evaluation)?

Svarsalternativ	Graf (%)	%	Σ
no		11	2
yes		89	16

TELL US WHAT WE SHOULD KNOW!

What in this course has been particularly good?

- Interesting lecturer. Lectures are fun too attend. ¹
- The topic is interesting. ²
- Doing the assignments was hard work but felt quite rewarding, as there was much to learn from them - not only about the quirks of GeCode/J but the more theoretical parts as well. Overall, they struck a good balance between theory and practice. The absolute highlight of the course for me was Régin's allDifferent propagator. That's a sexy algorithm if I ever saw one! I wouldn't mind more material on similarly interesting global constraints in the course, and I'm sure there's plenty of goodness out there. The bonus system was great. The main reward of doing the assignments was still what you learned and not the points you got, since scoring enough to get bonus points was pretty hard. But without the bonus system, it would be hard to motivate oneself to do assignment 2 after getting a reasonable score on the first one. ³
- It's good that we have a well mailing system so that people can give and answer questions openly. It's a good atmosphere and I like it. ⁴
- learning about principles of constraint technology. i enjoyed a lot experimenting with

different real life or puzzle problems. very steady schedule and timing of the course(Mondays & Wednesdays). sitting in Pierre lectures. ⁶

- The lectures were quite good. There were rich contents and they were delivered engagingly by the lecturer. ⁸
- The lectures are good. ⁹
- The assignments and the project were challenging but achievable, and all of them were interesting. Lectures were informative and engaging. Constraint programming is a very different approach from anything I'd done before, but now that I've seen it it seems like a technique I'll come back to often. In short, an excellent course. ¹⁰
- Interesting course in general. ¹¹
- The assignments were good. The lecture was good. ¹³
- We have been shown a very nice set of problem solving techniques and tools. Good assignments. ¹⁴
- The assignments and the project offered a stimulating challenge. The lectures were well structured and informative, but unfortunately there was not enough time to cover all the material. It was also nice to learn a new and powerful tool for problem solving. ¹⁵
- The slides are particularly good, because it shows the teacher's thoughtfulness. And the experiment design is pretty good as well. It has been a lot of fun. ¹⁶
- The lectures were focussed, clear, pedagogic and efficient. Overall, the lecturer gave his insights with high "pedagogic generosity". Also, the assignments and the project (not yet completed) overall gave good insights, with a small reservation given in next question. ¹⁷
- I think the course in general widened my perspective, therefore I really appreciate the last two lectures (concerning local search and comparing CT to other optimization technologies). It was interesting to realize that Régis's algorithm for all-different is exactly what I do when I solve sudoku, but much more general. I think the time spent solving the assignments was reasonable and it really helped me prepare for the exam. ¹⁸

How could the course be improved?

- Better lecture structure, too much jumping in slides back and forth. Slides or some notes should be more verbose. It is too difficult to work with study material, because it offers too little details. Sometimes felt lost during lecture, failing to understand how it is going on and why. On the other hand, sometimes lecturer was explaining kind of obvious things for extended amount of time. ¹
- As I'm sure you're already aware, the slides could be better structured. Trying to actually implement the assignments before posting them is a fool-proof way to avoid issues such as the GeCode/J bugs with ES_SUBSUMED and boolean branching. I appreciate that you are on a time budget, but so are the students - banging our heads against the wall dealing with bugs in the framework may be excellent practice for life as a software engineer, but in a 7.5 point course about something completely different it's a waste of time. That said though, the instructors did a good and timely job of coming up with on-the-fly solutions when these kinds of problems popped up. The mailing list was also a very useful tool for communicating these things. Maybe a brief look at other frameworks than GeCode/J might be interesting, seeing how GeCode compares to things like Choco or JaCop or whatever else is out there. It doesn't have to be a gargantuan part of the course, just something to give us a vague feel for their different strengths and weaknesses. Also, a couple of slides about how to do A* search (or something similar) would be an interesting addition to the slides about searching. ³

- It would be great to have an introduction lecture on basic and popular constraint techniques and what's going on in constraint field.⁴

- about lecturer: i wish we had more lectures about how to translate a model into implementation, we had a very few complete examples, even warehouse example only came to mathematic model while we could also know more about how to rewrite that mathematic model into a solver like Gecode/J, that could be very instructive. about lectures: sometimes skipping some lectures during the lectures made it difficult to review those slides and try to find and understand the connection. specially some slides that representing the Gecode/J programing were not matching with Gecode/J library and in most cases this added an overhead for making a meaning out of the lectures. about assistant: unfortunately i had no hope to get an answer from him, most of the time he could not understand me or he had no solution for my question, even if he got my question clearly he couldn't answer it because of his poor English. about assignments: i wish assignments could cover more variation of problems, so that we have the chance to work on more different small problems instead of putting too much time on one big and repetitive question like n-queens. about Gecode/J library: this is a good library for teaching but unfortunately it lack a lot of documentation and this put too much pressure on students to make it work the way they expect. for example i could make many different mathematic models for the problem, but i had no idea how they should be represented in Gecode/J code. and since the documentation was so poor it stopped me from experimenting more. later in the course i started using the Gecode(C++) library and i found that much constructive to work with, and i strongly recommend to use it for next year, especially knowing that Gecode/J is already discontinued by Gecode development team. about exam: exam was a little too long, every question took long time to finish, for those who are not very fast thinkers and writers (like me) it was a pain to see time is going on and there is no more time to even read the last question and knowing that I'm capable of solving it. i believe the exam could be less long but more calibrated to find out if students understand the main ideas during the course.⁶

- 1.The library we use,namely gecode/j.It is quite buggy and poorly documenteted,which severely retarded the progress of solving the assignments and the project. Many an hour had been spent in finding out what a constraint method means and how to use it.Once in assignment 2,when we wrote a program,we found that it didn't work as expected and then we spent another several hours only to figure out that it was a bug of gecode/j.This was a really frustrating experience. 2.The assignments.Some questions were poorly designed.For example, in assignment one,it wanted us to run the n-queen problem from n=1 to n=100 and for at least 8 combinations of variable and value choices.We set the timeout to 1 minute and after around n=20,it had the timeout all the time.Surely we can write the program and let it run there automatically.But one combination needed 100 minutes and there were 8 combinations,which means we need 800 minutes---roughly 13 hours just to finish one part of a question and within these 13 hours, we could not use our machines to do the work of other course. 3.The exam.I feel that the exam was too heavy.I consider myself as an active participant of this course.I solved the exam of last year before the exam.But when I sit for this year one, although I found I knew the knowledge points underlying the questions,when I got down to answering a question,it appeared that it would never end.An example is the question wanting us to draw a search tree.What I want to say here is that the lecturer really has to make sure that a question is really suitable for an exam and the instance data are fine tuned so that a student who understands the related knowledge can finish the question within an acceptable time,which is quite critical for an exam because it increases the confidence.Another example is the question asking about the bipartite graph.The instance data was so bad that the graph became very clutter in the second phase(oriented graph) and I

couldn't continue with it in the exam, which is really a sad thing in an exam. ⁸

- The bonus is difficult to get. The exam is difficult. ⁹
- More concrete examples of how to write mathematically correct models. ¹¹
- There should be more small assignments, to cover more subjects. The lecture should be more related to the practical use of gecode. I would have liked to learn more about the use and features of gecode. ¹³
- The textbook (Apt) has been more of a reference than a coursebook, so maybe it could be listed as such? The documentation of Gecode/J is very scattered (website, examples, slides and references). ¹⁴
- The sparse documentation of Gecode/J, in addition to the ES_SUBSUMED and BoolVar bugs, made some parts of the assignments unnecessary hard to solve. Of course, you cannot be held responsible for the documentation, but the bugs should have been discovered earlier. The number of slides could also be reduced to improve the readability. ¹⁵
- Maybe more discussion should be introduced. Work load can be a little bit higher. We felt the course to be rather easy to follow somehow. ¹⁶
- The lectures: Not the quality, hard to beat. But I think some more weight should be given to the modelling, even if I cannot suggest which other topic should pay the price. Particularly, the high level modelling given at the course as compared to the lower level demanded at the exam is a bit tough. More training on that is needed. The assignments: Assignment 1 gave us a lot of work with experiments, plotting and graphing, where much of this did not yield any insight, essentially because many of the different choices did not improve anything. Maybe the experiments could be shrunk down, and the choices of variable and value branches suggested in such a way that insight/work ratio was improved, without for that matter putting the answers in our mouths. Also, I still believe that allocating 2 potential bonus points to each of the assignments is better (even if it would not have helped us in the actual case, but who knows what we would have done with other rules, we handed in assignment 2 when we were sure that the min 12 pts on the project would be the only remaining constraint, the bonus giving one having already failed. The project: work in progress. Since this is about modelling, I think the exam should be after the deadline of the project. Exam: One day afterthought: I strongly feel that my disastrous performance does not reflect my beholding of the course, and that is good. I simply was not sufficiently prepared for the exam in the sense of having transferred what belongs to the fingertips down to them, which is done with training and solving equal problems, thus reserving the precious exam time for more productive thinking. I can blame nobody else for that. On the other hand, an exam is (at least for some) a stressful event, where nervousness can play you bad jokes. Therefore some comments: Q3: Was worth 30% of the whole exam, and it all relied on your getting the first little part (a) right. Even if this was a totally trivial model, I had the insecurity that it was `_too_ trivial`, that I had omitted some fine point, and that made me insecure and stressed about all following points, and I think it had an impact of my whole attitude towards the rest of the problems in that question, leading to... If the point is the ability to come up with some simpler model than the one in the fifth part, which I didn't even look into, maybe some clever choice of progression in Q5 could have shown such ability. As a suggestion, some easy modelling in initial questions, and the hard part later. Q3: This I really f...ed up, and I should have been able to show that I had understood the algorithm, since I still believe that so is the case. I thought that A and B were unrelated, and that the interesting observations were to come further down. I agree that looking at it the day after I should have understood that even Régine's algorithm :-) cannot do much about propagating 6 decision variables belonging to the same domain... I was confused about that and tried to make the best out of it with some

abrakadabra... That would have been eliminated if the beginning of statement B was: "Using the store obtained from A, propagate the constraint on the top left 3x3... etc". In an examtext, better to try to eliminate all ambiguities (I now suppose that immediately recognizing the absurdity of my approach thereby backtracking to the intended way was not part of the test, if it were, it is a dirty trick :-)) Best regards, Roland Hedayat (confidently awaiting for the next exam opportunity) ¹⁷

- I think some terms could have been explained better in the lecture notes (various levels of consistency), but usually the book answered my questions. I would appreciate to learn even more about global constraints. ¹⁸

Please be informative and constructive.

eValuering 2.1 Copyright © 2001-2007 Tobias Jakobsson.
Licensierat till Institutionen för Informationsteknologi, Uppsala Universitet.