Department of Mathematics and Computer Science
University of Southern Denmark, Odense

November 20, 2009
Marco Chiarandini

DM811 - Heuristics for Combinatorial Optimization

Handling Plan, Fall 2009

There were 20 students enrolled in the course and 19 registered to the exam. Of these, 14 submitted a project and 11 passed the exam. There were 11 students that filled the course evaluation form and 9 who did not. Separately, I received feedback via BlackBoard forum from 5 students. Students were asked to describe briefly the course and to express their opinions on the variation of activities, their preparation to the exam, the modalities of feedback and the content of the course.

The general impression remains that students experience the course quite differently. *Describing* the course, students are aware that a lot of knowledge from previous courses and a lot of programming is required. They recognize the practical drive. Some say that it is a course taught on examples, where no rules are given and hence there is more space left to creativity than in other courses, this leads to a different way of thinking that may make the course hard. A few comment that the course has a heavy workload and that this came unexpected hence it demotivated them.

The students find that there was a good use of different means in the run of the course, blackboard, slides, assignments and e-learning system (BlackBoard). Almost all mention to have experienced *variation* of activities during the classes, although it does not arise clearly from the comments whether this was always perceived positively. Most liked and found useful the running example with a toy problem although they all felt that something should be improved in that context.

The students feel *prepared* enough for the exam but mention that the hardest part is the design of the heuristics because the content of the course is wide and there are many possibilities without clue on what will work best. Some would have liked more training on the details and tricks with hands on examples.

As far as *feedback* to the students is concerned, they agree on the relevance and contribution to this aim of the assignments, both the programming part on the toy problem and the questions posed separately (which may help to understand the theories presented). They suggest that the assignments are made compulsory and the workload in the final project reduced. The task of keeping a lecture journal was not successful, a better way would have been to pose questions regarding the content of the lectures. A student suggests to not use slides in class and focus more on student-teacher interaction. In this regard, students should not be answered questions directly if they did not raise their hand and there should not be reprehension for unanswered questions. Also, there should not be overlapping weekly assignments of different types. The same student suggests as improvements easier and well paced lecture notes and aforehand decision on which lecture will be based on assignments. Finally, more detail on the specific heuristics is asked with a run example and close explanation of every algorithmic iteration.

Concerning the *content* of the course, the comments are mostly positive and few have suggestions on what could be removed or what should be treated deeper. A couple of students point out that some parts are overlapping with other courses, like the reducibility from SAT and basic algorithmic concepts or data structures. It would be

instead appreciated if more time was invested on the implementation of local search with closer examples.

From this reflections, the following concrete actions will be taken in planning the next edition of the course:

- Introduce obligatory assignments pass/fail during the course that include programming while still maintaining the final project with 7 scale grade. This requires to change the course description and have it approved by the student committee.

- Base the course on the system COMET that will facilitate the programming part while having just a little initial learning barrier.

- Remove the lecture 3, that is, the resume on basic algorithmic concepts, definition of NP-hardness and problem transformations.

- I will try to give some guidelines based on empirical results to help students in the design process.

From the exam projects I noted that some mistakes I would like to see avoided where instead present. Among them: non competitive local search procedures and mistaken data aggregation in instance set analysis. I will insist more on these issues, possibly removing more advanced topics like very large scale neighborhoods that are however not relevant for the exam.

Finally, an anecdote: when the final grades where communicated, students where told that they could come in my office to receive the written comments and grade justifications. From 14 students that handed in the exam, only three came and of them two were those who got the highest grade. The BlackBoard system does not allow at the moment to add a comment to grades in the grade book.