Investigations
in university teaching and learning vol 2 (1) summer 2004 ISSN 1740-5106

Improving Teaching Effectiveness by Grading Curricular Content: a case study from Computing

Na Helian
Department of Computing, Communications Technology & Mathematics
London Metropolitan University

Key words: *graded tasks; scaffolding of learning;, large class teaching; student diversity*

Introduction

With class sizes in British higher education now approaching what is common in mass education systems, the range of abilities and background in a class are becoming more varied. If we care about quality but cannot increase resources then we have no option but to change educational methods (as Jenkins and Davy (2002) argue). This paper discusses changes made in a computing module that were aimed to ensure effective and efficient teaching.

Module Background

Data Modelling and Database Systems (IM207) is an advanced BSc module at department of computing in London Metropolitan University (North Campus). Databases today are essential to every business. They are used to maintain internal records, to present data to customer and client on the World Wide Web, and support many other commercial processes. Databases are likewise found at the core of many scientific investigations, such as gathering data by astronomers, exploring the medicinal proteins by biochemists, etc. Data modelling and database design are among the highly demanded skills in industry and scientific research and viewed as essential components of computer science programs, as well as many computer application programs.

The module is delivered using a combination of lectures (2 hours) and laboratory/tutorial sessions (2 hours). Generally, the students should spend equivalent time (4 hours) on their self-study for each session. Assessment is by coursework and examination, with each component equally weighted. To pass the module it is necessary to pass both components. The coursework is organized as an individual project, while the exam is an unseen, closed-book exam with a quantitative orientation. The examination tests the students' retention, understanding and insight of material drawn from the entire course. In an individual coursework (building a substantial database application for a real-world scenario of each student's choosing), students are asked to enrich their work gradually after each lecture by adding up implementation of the new theories they have gained in the lecture each week.

At the time of this study (2002/03), IM207 was a large class comprising around 160 students from a variety of disciplinary backgrounds. It was a heterogeneous mix: some were 'traditional entry' students straight from school; the remainder had radically different qualifications and experiences. There was certain number of mature students returning to full or part-time study. We also had a high percentage of international students. While most students on this module had almost no knowledge on database concepts, theories and implementation, a small portion, mainly 'traditional entry' and international students, had some preliminary knowledge or

experience. It was very important to respect diverse talents, and help all students engage with the subject, by make the knowledge tangible to everybody. This meant that the module should not be too difficult for the students without any knowledge, but interesting for those who had certain knowledge on the subject.

For the large class, it is unlikely for the tutors to respond flexibly to students through personal contact. So how could we cope with the problem? Grading the curriculum content in terms of difficulty and complexity seemed a feasible way, plus overlapping the essential and difficult topics in different sessions.

Learning theory and the new teaching strategy

The "zone of proximal development" is the Vygotskian concept that defines development as the space between the level of independent performance and the level of maximally assisted performance. Proper assistance should increase the students' level of performance beyond what the students may achieve on their own, or with instruction that is out of their range of capabilities. The term "scaffolding" was coined by Bruner; a suitable "scaffolding" provided by a lecturer should make it possible for a learner to complete the task with support, rather than make the task too easy or difficult (Bodrova and Leong, 1998; McKenzie 1999). Hence, according to these concepts, grading the content of the module in terms of difficulty and complexity could help all students to achieve their maximum performance.

Instruction is only useful when it moves ahead of development, so we should offer step-by-step direction to explain what the students must do in order to meet the expectations for their learning activities. The module aimed to facilitate students' learning from dependent towards autonomous learning through the process of going from introduction of new concepts and theories, to demonstration of the applications, then to different level exercises/tasks, and ultimately self-selected projects.

In the module's original teaching regime, students were being treated as a homogeneous group. The module content was geared more or less towards the 'fast' students. For each session, usually after introducing a new concept, a demonstration of implementation of the concept would be conducted. There were very few in-class exercises for students. It was reflected in the exam that just the 'fast' students (roughly 10%) could solve the problems by their own on difficult topics. Although students said that they understood what they had learnt in class, due to the lack of in-class exercises and tutor feedback, students lost this crucial sense of how they were doing. Our aims included an intention to make the module as flexible as possible to cope with the diverse demands of students with very different academic background. So we had proposed a new teaching strategy on the module that involved introducing in-class exercises of differing depth and complexity (i.e. grading the content), each to be followed by specific explanations.

As these exercises gradually increase in difficulty, this could develop the 'slow' students' understanding of a particular topic step-by-step. On the other hand, there are more difficult complex exercises provided, so the 'fast' students' knowledge could be widened. Different students with different needs could take advantage of these exercises to different extents. This matches the individual student's aspiration and allows students to reach targets that are suitable for their level of progress. By doing the tasks, students know whether they are understanding things adequately and generally keeping up. In addition, the in-class exercises are intended to encourage active learning and allow students to learn from doing.

Implementation

The changes made in teaching the module focused on the two topics (decomposing a many-to-many relationship in an entity relation diagram, and self-join query) on which students had appeared weak in the previous semesters. For example, in session 2, after introducing the concept of data modeling and database design theories followed by a demonstration on the decomposition process of a m:m relationship as usual, students were asked to do exercise 1. As expected, just a small percentage (less than 10%) of students could do it correctly, while most made mistakes. Next, the correct solution was demonstrated. Then, the mistake, which most frequently appeared in their solutions, was pointed out and the reason for that was analyzed. Following this, exercise 2 was released. Although even fewer students finished it properly, it was still a good sign that most students could at least start doing it. When the similar questions had appeared in previous semester students' coursework, they got totally stuck. During the tutorial for session 2, most students did very well on the exercise which was on the same level of difficulty as exercise 1 given in the lecture. With regard to the case study used for session 2, although the majority of students had difficulty with it, this exercise allowed students to know where they were weak and enabled them pay appropriate attention when the solution was interpreted in session 3. As a matter of fact, even before session 3, most students asked for the specific explanation on the case study, which indicated that they were ready to learn the topic more deeply.

Evaluation

While delivering the module, my concern was how well the new teaching strategy worked. Thus, I collected information by informal interviews with students and discussed the changes with the module's teaching assistants. I also got feedback on this from teaching observations by colleagues.

Feedback from the students
The feedback from the students on the in-class exercises with different levels was satisfactory and encouraging. The students who did not have any pre-knowledge on database felt that they did gain knowledge on database design and implementation, and the in-class exercises helped them understand the theories and techniques. Those who had some preliminary knowledge on database said the module was a systematic and logical recapitulation of what they had learnt before; it enhanced their understanding and extended their knowledge on database. Furthermore, the different levels of exercise enabled them to deal with more complex applications in the real world.

Students' assessment results
The number of students passing the module increased by nearly 6% from 75.8% to 81.5% (compared with the previous semester) and the average mark increased from 50% to 55%. There was also a welcome increase in the percentage of A and B grades, from 35% to 40%. The students in these grade levels produced good quality coursework based on their own personal database application scenarios, and did well in the unseen exam. Notably, 90% of students could solve the problems on decomposing a m:m relationship, and 38% of them could make a basic self-join query correctly, which was a dramatic increase compared with the previous semester. These were the very difficult topics to which the new teaching strategy had been applied.

Peer feedback
The teaching assistants did think this was a good way to enhance students' understanding and meet different levels of students' needs. After the teaching observations, peers said that the in-

class exercises could make students more involved in the session and promote their active learning. One observer noted that interspersing the tasks alongside the lecture presentation helped to involve the students and to contextualise the content. Based on such advice, I was encouraged to explore ways of varying the lecture (e.g. by use of small student-based activities) and to consider greater use of case study materials to improve students' engagement.

Conclusion

To meet different students' needs in a large group with various academic backgrounds, a new teaching strategy was tried out, that involved grading the module content in terms of difficulty and complexity, and overlapping the essential and difficult topics in different sessions. After implementing the new teaching strategy on this module, we had got positive comments and views from students, tutors and other colleagues. The in-class exercises made the lecture more participatory, and prompted students to take a more active part in class. According to the students' assessment results, we improved the pass rate and average mark by implementing the new teaching strategy. Therefore, we can conclude that grading teaching content is a feasible and effective way to deliver teaching and facilitate learning in a large class with a diverse student body. While observing the students, it also seemed to me that they were gaining not only technical knowledge, but also learning skills which are transferable to other modules and would benefit their future careers.

References

Jenkins,T and Davy, J (2002), Diversity and Motivation in Introductory Programming, *Innovation in Teaching And Learning in Information and Computer Science*, Volume1, Issue1, http://www.ics.ltsn.ac.uk/pub/italics/issue1/tjenkins/003.html

Bodrova, E and Leong, DJ (1998) Scaffolding Emergent Writing in the Zone of Proximal Development, *Literacy Teaching and Learning*, Volume 3, No. 2

McKenzie, J (1999), Scaffolding for success, From Now On; *The Educational Technology Journal*, Volume 9, No. 4

Biographical note

Na Helian received her Ph.D in computer science in 1993. After 8 year's research in the computer field in China, Japan, Singapore and UK, she joined the London Metropolitan University as a lecturer in 2002, in the Department of Computing, Communications Technology and Mathematics.
Contact: n.helian@londonmet.ac.uk

Acknowledgements: the author would like to acknowledge the input of Dr Frank Wang, Reader in Grid Computing in the Department of Computing, Communication Technology and Mathematics, and Digby Warren, the editor and a Learning, Teaching and Curriculum Development Coordinator in the Centre for Academic Professional Development.