Improving teaching in *Computer Programming* by adopting student-centred learning strategies

Hu Xiaohui
Computer Center
Ocean University of China
Qingdao 266003
People's Republic of China

ouqdchina@yahoo.com.cn

Abstract

Currently, the teacher-centred teaching and learning strategies still dominate the classrooms of most of universities in China. In this paper, new strategies to improve teaching and learning quality, are introduced. First, current conditions and characteristics of *Computer Programming* in Ocean University of China (OUC) are introduced; then the advantages and disadvantages of the traditional teacher-centred learning strategies are discussed. After that reasonable modifications from a strongly teacher-centred approach of teaching and learning to a more student-centred approach of teaching and learning are suggested, based on the author's teaching experience. Some modern student-centred strategies and techniques including concept mapping, peer learning methods and e-learning are introduced and their applications in the course of *Computer Programming* are discussed.

Introduction

With the development of science and technology, people are facing a rapidly changing knowledge based society, and hence, good communication skills, flexibility and adaptability are required to continue learning throughout their lives. As teachers, our most important task is to cultivate students with these abilities so that they can acclimatise themselves to a fast developing society. Therefore we need to utilise appropriate teaching and learning strategies to enhance a deep level learning, activate students to construct knowledge, and cultivate the lifelong learning skills.

In semester 2, 2006, as visiting scholars in The University of Sydney, we have had the opportunity to join the *Teaching Science in English* program. This program is designed to introduce the contemporary issues and research in the teaching and learning of science. From the systematic courses presentations and a series of fascinating seminars, we investigate the impacts and implications of contemporary research findings and have a better understanding of them.

According to Kember's (1997) description, there are two kinds of broad orientations in teaching: the teacher-oriented conception and the student-oriented conception.

The teacher-centred learning strategies are described as focusing on the teacher transmitting knowledge from the expert to the novice. In contrast, the student-oriented learning strategies are to focus on the students' learning and 'what students do to achieve this, rather than what the teacher does' (Harden and Crosby 2000).

Teacher-centred learning strategies

The traditional teacher-centred learning strategies include such activities as lecturing, questioning and demonstration. These strategies have dominated classrooms all over the world for a long period, especially in China because of the effectiveness of Confucianism.

There are some advantages about the teacher-centred learning approaches. For example, they are efficient and effective for lower-level learning, especially in large groups. They may provide foundational information preceding other methods.

In this information society, the traditional teacher-centred learning strategies have been found to be inadequate for many students. The teachers focus on what they want the students to learn and restrict what the students might learn. That means teachers put more emphasis on the knowledge itself rather than developing

Mentor
Bing Bing Zhou
School of Information Technologies
Faculty of Science
The University of Sydney

students' learning skills. Therefore students are passive learners during the learning process. They tend to memorise the knowledge packages taught and recall them on the examination paper. Instead, they cannot grasp and assimilate the knowledge effectively by themselves. The traditional transmissive teaching methods lead to superficial learning of science and the persistence of student misconceptions in most cases.

How to make students active seekers instead of passive recipients of information and knowledge is a challenging and essential issue in the current teaching and leaning scenario. A shift from a strongly teacher-centred approach to a more student-centred approach may provide a potential solution to this challenge.

Student-centred learning strategies

In the student-centred approach, it is believed that knowledge is constructed by students and the lecturer is only a facilitator of learning rather than a presenter of information.

In this way, students become more active participants in the learning process wherever possible and take greater responsibility for their own learning. These strategies encourage deep level processing of information, make the students efficient problem solvers, and develop lifelong learning skills.

In this paper some of these student-centred learning strategies will be introduced; the difference of teaching methods between the author's university and the new teaching methods learnt at The University of Sydney will be discussed, and some suggestion on modifying the teaching methods in the author's course will be given.

Current situation

OUC is a comprehensive university which offers courses in Science, Engineering and Economics etc. The curricula of Information Technology in OUC are designed for those students who do not major in computer science. It has three layers: the *Fundamental of Information Technology*, the *Computer Programming*, and the *Advanced Information Technology*. Each of them is a one-semester course with three credit points.

The *Fundamental of Information Technology* is the prerequisite course for the *Computer Programming*. Both of them are compulsory courses. The *Computer Programming* course has two-hour lectures in the classroom and two-hour practices in the computer laboratory per week. We give students a list of computer programming languages. They can choose one of them according to their interests and disciplines. Because the number of students in each class is at least 80, we still adopt the teacher-centred approaches as our dominant teaching methods. In the classroom, students are given lectures and program demonstrations. In the laboratory, students ask questions and are guided to finish their assignments.

All the students are expected to attend two closed-book examinations (mid-term and final term) and one online programming practical examination. Their forms and weights are listed in Table 1.

Table 1.

	Weight (percentage)	**Form**
Mid-term examination	10	close-book, paper
Final term examination	60	close-book, paper
Practical examination	20	online
Other	10	assignments and attendance

I have been teaching this course for several years. Like most of the other teachers, I tend to teach in the way I myself learn best. Although I try my best to teach them the knowledge of my course, I find what the students learn is not a mirror image of what is being presented by me. Only those students who are interested in this course can self direct their own studies, learn quickly and apply what they learn to solve real world problems. While others who consider this course a compulsory task, only finish what I have assigned. The problem with this group is that they may complete the assignments while totally missing the concepts behind them.

Contemporary teaching approaches and their applications

Through the study at The University of Sydney, we were introduced to ideas about 'contemporary theories of teaching and learning' in science and came to understand the trends that are occurring in Western universities. I realise active learning strategies which engage students in discovery and scientific process will improve learning and retention of knowledge.

Actually, it is hard to say which kind of learning strategy is totally correct or wrong. We shouldn't abandon the traditional teacher-centred strategies completely. It is perhaps better to make modifications to the traditional teaching methods according to new ideas and methods because evolution is better than revolution.

Due the reasons mentioned, some contemporary teaching methods will be introduced in this section, and then their application to the course of *Computer Programming* based on my own teaching experience will be discussed. These modern incorporating strategies include concept mapping, peer learning methods and e-learning, etc.

Concept mapping

Concept mapping is a technique used to visualise the relationships between different concepts graphically. Knowledge graphs are networks of related concepts and ideas that are interconnected. These graphs consist of nodes which represent related concepts within a topic and links

which represent the relationship between concepts (Lanzing 1997).

Concept mapping can be used to generate ideas, to communicate complex ideas, to aid learning by explicitly and graphically integrating new knowledge with existing knowledge, to assess understanding and diagnose misunderstandings about knowledge.

In the course of *Computer Programming*, I used to give the students a brief course outline at the beginning and a summary at the end of the semester. Usually I choose the form of text with tables and lists. At the end of semester, students may only get pieces of knowledge. They couldn't generate a clear and total concept themselves. When I came to understand the technique of concept mapping in the class of *Teaching Science in English*, I decided to adopt this technique in my teaching.

At the beginning of each semester, I will give students a brief concept map of this course (Figure 1).

This concept map looks like a tree in winter with only some main branches. It can serve as a key plan for me in helping students retain what computer programming is and what they will learn in this course.

As time goes, the leaves will grow. When we finish one chapter, I will guide students to draw their own concept maps and compare their maps with mine.

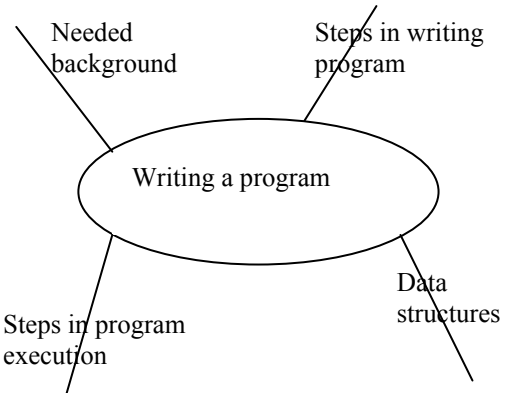

Figure 1. The concept map at the beginning

Figure 2 shows a rough concept map of *Computer Programming*. At the end of each semester, I will give them the integrated map and ask them to give me their own concept maps of this course content.

This can help me establish what they have learned and what they still do not understand. The students can integrate the pieces of knowledge into a clear concept map. They will also know which part they should pay more attention to.

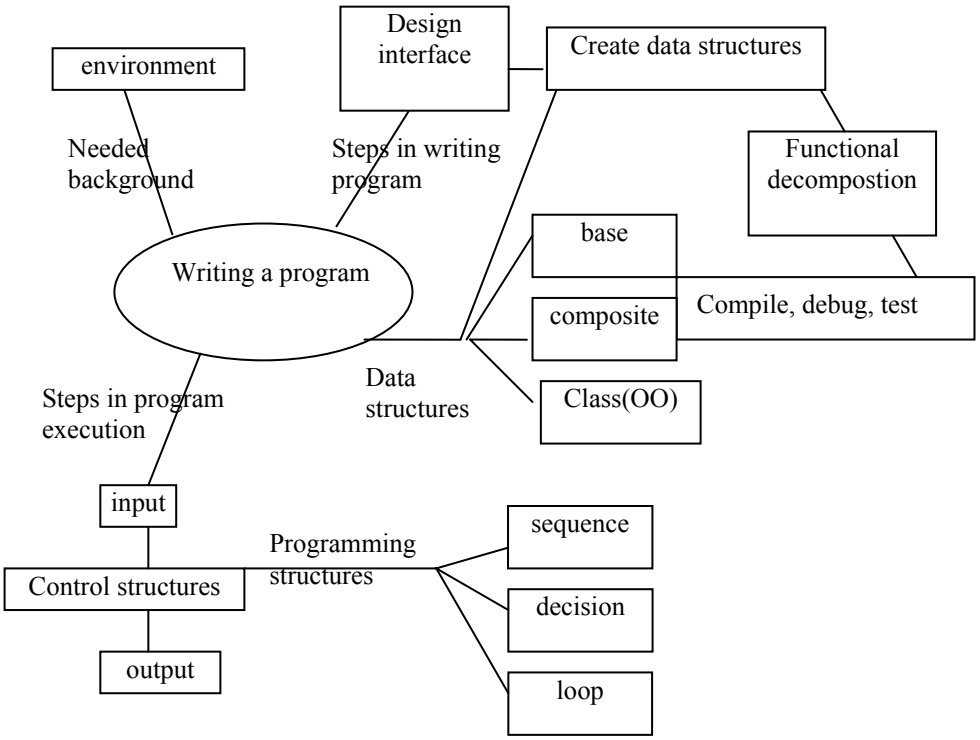

Figure 2. A rough concept map of *Computer Programming*

Peer learning methods

Learning is essentially a human activity which is cooperative rather than competitive. People do not learn things best at the expense of others. They need to share with their peers and validate by the whole learning environment. I will discuss two kinds of peer learning methods and their applications to my course. One is group work, the other is peer tutoring.

Group work

Teachers should use as much group work as they can as well as individual tasks because knowledge is 'socially constructed'. Through group work, students can apply skills and knowledge learned elsewhere to solving a specific problem. They learn to cooperate with others and gain specific knowledge and skills which are very useful for their future work in industry.

During my teaching in *Computer Programming*, I only give my students individual compulsory tasks so that I can give them practical marks according to their results. Besides that, I also give them additional projects which students can choose to finish or not. Those who are interested can communicate with me. Extra bonus mark will be given according to what is handed in. At this stage, I have not trialled group work with my students.

At The University of Sydney, many courses have group work as well as individual tasks. The group tasks are designed to help students learn more effectively. I am considering developing group work in my course. It will encourage the students' active participation in group discussions of topics covered in class. Learners will get the opportunity to articulate their beliefs and thoughts on the topic to their peers.

All of the group members will obtain teamwork skills such as communication skills and interpersonal skills. The group leader will also develop leadership and organisational skills.

Peer tutoring

Peer tutoring is the system of instruction in which learners help each other and learn by teaching (Goodlad and Hirst 1989). It is beneficial in helping students learn actively in a setting that promotes multiple-intelligences (Gardner 1993). It is one of the most effective methods in promoting student-centred learning.

Student tutoring is beneficial to all the groups including tutees, teachers and tutors themselves.

The students would receive individual attention and so learn more than usual. They would overcome shyness, ask questions and be more involved with activities since tutors and students are peers.

The tutors learn to teach students working at their level and ability. They will increase their communication and interpersonal skills too. They also reinforce their knowledge of the fundamentals of this subject while tutoring others and receive practice in problem solving and time management.

The University of Sydney uses student tutors in many disciplines. Most of the student tutors are honours or PhD students who must have a good knowledge in the courses they tutor. I have attended several tutorials guided by them. Teaching by the student tutors comprises a short presentation followed by a class activity led by the tutors. The teachers' main task is to facilitate the activity of the student-tutors as they teach their tutees. The student tutors will discuss the teaching content for each tutorial with the teachers.

According to the actual curriculum design of the *Computer Programming* course in my university, we do not have special hours allocated for tutorials. When considering peer tutoring, I would suggest introducing student tutors in the laboratory. They would be outstanding students from a previous or current semester. When completing their assignments, the students could have discussions with their tutors in and after class.

E-learning

E-learning is an all-encompassing term generally used to refer to computer-enhanced learning (Wikipedia). It may include the use of web-based teaching materials and hypermedia in general, such as web sites, discussion boards, wikis, computer aided assessment and more, with the possibly of a combination of different methods being used.

In higher education, the discipline of Information Technology is the most appropriate one in which to use e-learning. The reason is that the technology used by e-learning belongs to Information Technology.

The online course environment

From my experience in teaching *Computer Programming*, I found it quite important to build a course learning web site for my students. In fact, I have designed a web site and used it for several semesters. The course web site has many advantages such as students can:

- download the lecture notes, preview and review them after class, so that this will save them time in class and let them know which part they should put more emphasis on; and
- access the relevant resources and upload their assignments through the Internet.

The communication between students and teachers becomes more convenient. Through the discussion board, I can answer students' questions after class even if we are in different places.

Although my web site has had a good effect on my students' learning, I still found some problems with it. First I have to find a public web server where the source code of the web site can run. The web server should work 24 hours a day/seven days a week so that the students can access the learning resources. Second I not only have to update the material and answer questions, but also maintain the web site technically. If my major was not in computer science, it would be impossible to design and maintain this web site myself.

Now I think I have found a solution to this problem. As e-learning has matured, many universities adopt such useful tools as *WebCT* (Web Course Tools) or *BlackBoard* to solve these problems in Western countries. The University of Sydney adopts *WebCT* as an online course platform.

The course management system *WebCT* was created by an educator and nurtured by a community of educators to become a powerful tool in use by instructors today. It comprises an integrated set of educational tools which can facilitate instructors to construct and manage an online course environment.

These tools include course contents publishing, course progress tracking, course calendar, forum, online quiz and chat room etc.

Compared to a personal web site, *WebCT* has advantages such as:

- teachers in other majors can also build their course websites even if they know little about information technology;
- freed from the maintenance of web sites technically, teachers can put more emphasis on teaching; and
- although it may not be as elegant as personal web sites; this platform has a uniform interface so that students can easily learn how to use it. Sometimes simple is the best.

The utilisation of Version Control System

I have discussed why teachers should use group work as well as individual tasks before. Considering my major and course, most of the group work in Information Technology are design projects or writing programs in my future teaching. They need communication and discussion. The group should have a good leader and diligent members.

While implementing the group task, more than one group member may be editing the same program file. In this occasion, the most important thing is to control the software's version. One way to help assure quality and version is to monitor progress carefully. Software such as *Wiki* and *Trac* can implement this function.

In Cunningham's (2002) original description: '*Wiki* is the simplest online database that could possibly work'. It 'is a piece of server software that allows users to freely create and edit web page content using any web browser'. '*Wiki* is unusual among group communication mechanisms in that it allows the organization of contributions to be edited in addition to the content itself.'

Trac is the system used by the school of Information Technologies at The University of Sydney. *Trac* is an enhanced wiki and issue tracking system for software development projects. *Trac* uses a minimalistic approach to web-based software project management. It aims to help developers write great software while staying out of the way.

This kind of software hasn't been widely used in Chinese Universities. I would like to introduce them to the course of *Computer Programming* in my university. *Trac* makes it possible to register all activities in order to figure out who's doing what in a group task. It can also motivate the students to work in a constructive way. Students can cultivate the ability of working in a group which is very useful for them for the rest of their lives.

Summary

From the study, investigation, and involvement in this program, I have learnt the difference between teaching in China and in Australia. In this paper, some student-centred strategies such as concept mapping, peer learning methods and e-learning are discussed. There are still many other useful strategies, which can develop students' learning ability and skills, such as case study, problem-based learning etc, have not been mentioned. Considering the actual conditions in China, we'd better choose the combination of the traditional teaching approaches with some new approaches. What I have learnt here will have a profound impact on my teaching in the future.

Acknowledgements

I greatly appreciate the China Scholarship Council and Ocean University of China for their support and giving me this opportunity. I would also like to thank The University of Sydney, Faculty of Education and Social Work, Centre for English Teaching, Faculty of Science and School of Information Technologies. I wish to give grateful thanks to Associate Professor Mike King, Dr Tom Hubble and those teachers who have given their wonderful lectures. Special thanks to my mentor Dr Bing Bing Zhou for his help and valuable discussions. Many thanks to the teachers in this program for their lectures, their help and encouragement. Thanks to my classmates for making life in Sydney enjoyable!

References

Angelo, T.A. (1991) Ten easy pieces: Assessing higher learning in four dimensions. In T.A. Angelo, (Ed.), *New Directions for Teaching and Learning: Classroom research: early lessons from success*, **46**, 17–34. San Francisco: Jossey-Bass Publishers.

Cunningham, W. (2002) What Is Wiki? [Online] Available: http://www.wiki.org/wiki.cgi?WhatIsWiki.

Duff-Riddell, W.R. *On Trac in science education*. [Online] Available: http://www.saasta.ac.za/scicom/pcst7/duff-riddell.pdf .

Edgewall Software. *The Trac Wiki Engine* [Online] Available: http://trac.edgewall.org/wiki/TracWiki.

Gardner, H. (1993) *Multiple intelligences: the theory in practice*. New York, NY:Basic Books.

Goodlad, S. and Hirst, B. (1989) *Peer tutoring: a guide to learning by teaching* London:Kogan.

Harden, R.M. and Crosby, J. (2000) AMEE Guide No 20: The good teacher is more than a lecturer-the twelve roles of the teacher. *Medical Teacher,* **22**(4), 334–347.

Kember, D. (1997) A reconceptualisation of the research into university academics conceptions of teaching. *Learning and Instruction,* **7**(3), 255–275.

King, M. (2006) Lecture Notes: Teaching Science in English, The University of Sydney.

O'Neill, G. and McMahon, T. (1995) Student-centred learning: What does it mean for students and lecturers? In G. O'Neill, S. Moore, S and B. McMullin. (Eds) *Emerging Issues in the Practice of University Learning and Teaching*. Dublin:AISHE.

Tan, C.H.P. (2003) *Peer Tutoring—An Effective Strategy to Promote Student-centred Learning* [Online] Available: http://www.cdtl.nus.edu.sg/link/nov2003/learn2.htm.

Townsend, S. and Fetters, P. (2004) *Teaching science the way you do in science* [Online] Available: http://www.hhmi.org/news/042304.html.

Wieman, C. (2006) *Using the tools of science to teach science*. University of Colorado [Online] Available: http://www.aas.org/publications/baas/v37n4/aas207/895.htm.

Wikipedia [Online] Available: http://en.wikipedia.org/wiki.