

A Case Study in Teaching Adult Students Computer Programming

Min Hu
Tairawhiti Polytechnic
Gisborne, NZ
minjie@tairawhiti.ac.nz

ABSTRACT

This paper introduces a case study method to teach adult students computer programming at tertiary level. It applies adult teaching theory to computer programming education. The author is pleased to share the experiences and findings with other teachers who also teach computer programming at this level.

Keywords

Computer Programming, CBC and Adult Students

So, both individual and group assignment can meet most students' cultural background.

1. INTRODUCTION

Teaching computer programming is always thought as the difficult part of any computer course. However, we are not only training students how to use pre-made programmes but also teaching them how to create new programmes for other people to use.

Programming is one of the important modules in the Certificate in Business Computing (CBC) course. I have inherited the teaching of it from a previous tutor who had used C++ language at Tairawhiti Polytechnic since 1997.

2. BACKGROUND INFORMATION

Most students at Tairawhiti Polytechnic are adult learners. They come from different cultural backgrounds with different learning styles. Therefore, different teaching methods have to be applied especially in teaching programming.

2.1 Cultures

Collectivism is common in Asia, Africa, South American, and the Pacific (Lusty 1993). On the other hand, individualism is a cultural pattern found in most Europeans (Trandis, H. et al.1988).

2.2. Learning styles

Some students are very visual. Others are more auditory types. Others are kinaesthetic: they have to stand and move (O'Connor 1999). The strategy is to use VKA (Visual, Kinaesthetic, Auditory), all three ways together in the class. It follows that it is best to:

- ◆ work in a visual programming environment;
- ◆ graphically present logic in chart;
- ◆ use Data show, OHP and handouts;
- ◆ teach in computer room;
- ◆ provide sample results and electronic handouts;
- ◆ finally, give practical assessments.

2.3 Survey

A survey is given to students who have already completed CBC course. It gives feedback on how they feel about these strategies.

3. LITERATURE REVIEW

In terms of which computer language is best for tertiary education, Levy (1995) reported that since the 80s, many schools had turned to C and C++ from Pascal in their introductory courses. Microsoft Corporation (1998) also reported that 55% of developers use Visual Basic (VB) as their primary development tool, compared to 19% who use C++ and 5% use Java.

Knowlton (2002) discovered that VB is a good language to prepare students for C++. However, Nelson and Rice (2000) argued that many students struggled

with VB in the introductory course, but then get on quite well with C++ in the advanced course.

Levy (1995) concluded that it doesn't matter which language you choose as long as it has an emphasis on problem solving, algorithm and logical thinking; not just on the advanced features of a particular language.

When choosing a textbook, people may come across a huge variation in the time it takes to teach programming from 24 hours to 21 days. Norvig (2001) argued that this is too big a rush and that programming is not somehow fabulously easier to learn than anything else.

With regard to methods of teaching programming, a lot of people find it difficult to teach programming because of its reputation for being "hard". Jenkins (2001) argued that a teacher's main task is to motivate students to learn. But it is too general and does not cover the details of how to teach programming.

Fincher (1999) summarised four approaches to teaching programming. However, it is a high level approach for setting the course. More detailed methods of teaching programming are discussed in this paper.

4. HOW DO WE DO

The following methods are used for teaching programming at Tairawhiti Polytechnic. Survey results provide feedback as to how successful these methods were.

4.1 Choose a basic computer language as the first computer language

It is hard for some students to learn C++ as their first computer language. During all the years of my teaching, I have put a great deal of effort into making this module easier for students. Finally, VB is the best choice.

4.1.1 Step 1. From DOS version to Windows version

In my first year teaching, I inherited command-based DOS version Turbo C++. Students needed to remember not only computer language but also command to operation. Next year I turned to use Visual C++ 1.0 and then 5.0.

4.1.2 Step 2. From C++ to Visual

Basic

I introduced VB to what is now the Certificate in Introductory Computing instead of QBasic in 1999. Next year, VB was introduced to CBC instead of C++.

The moderation for the year 2000 showed that this choice was similar to the one used by other polytechnics. There were 1/3 of the institutes using C++, 1/3 using VB, 1/3 using other languages.

4.2 Choose a beginner's textbook with step-by-step instructions

There are many books about VB language and programming. A textbook should first of all, be easy for students to understand. It needs to provide written or oral exercises for students to gain an understanding of the concepts and syntax. Most important are the examples used in the book. These examples should have step-by-step instructions for students to follow, rather than only providing a list of code. It should also have certain practical exercises to enable students to apply what they have learned. At the end, answers should be also included for self-checking. A compiler of computer language is a bonus in the textbook. It makes it possible for students to do more exercises at home rather than only on campus. Schneider's (1999) books have been used at Tairawhiti Polytechnic for three years.

4.3 Take more time than other modules

Normally, there are seven weeks for each module of computing courses at Tairawhiti Polytechnic. It always means that students are busy at tests and assignments at the end of each seven weeks.

The programming module has been extended to 12 weeks. It also relieves the stress from other modules.

4.4 Teach in the computer lab rather than in the classroom

"Seeing is believing". It is easier for adult student to understand what they learn if they are practising it in a computer lab. "They hear, they forget. They see, they believe. They do, they remember." Demonstrations with the Data-show are an advantage so students can adapt to a programming environment.

Also, keep all handouts on the Network. Then it does not matter if the students attend class. At the same time, it helps some advanced students to study more in depth.

4.5 Give a guideline to start

Students do not know what they need to do first, or next. Steps for students to complete each programme are recommended as follows:

- ◆ Analyse input, process and output
- ◆ Draw a flowchart
- ◆ Desk-check
- ◆ Convert flowchart to code
- ◆ Debug for syntax error
- ◆ Compare running results to desk-check to find out logic error
- ◆ Write external documentation

4.6 Graphically present programming logic

Some students like to code a programme directly. Simple questions are no problem. But nested selection and loop will make them dizzy. One of the solutions is to let students present the programming logic in a flowchart. It does seem to take extra time. Once it has been done it saves lots of time in modifying code. Conversion between flowchart and code also helps them understand programming logic from a different point of view.

4.7 Have a group assignment instead of only individual assignment

Each member of a group needs to complete one to two functions or procedures. At the end, the group leader will put all parts together to work in one programme. Students learn how to co-operate. They also learn how the data communicates and transfers between functions and procedures. Meanwhile, they also have experience how different part of a well-structured programme works independently.

4.8 Have practical assessment

Practical assessment not only assesses what students have learned, but also finds out how they applying their learning. Students can use different ways to complete their assignments. It is successful only if it works and meets the requirements of the syllabus.

Give students an open-book practical test rather than have them sit a closed-book written test. If students understand what they have done for each assignment, they can simply adapt their understanding

to the practical test. Anyway, it is easy to judge who is able to make a programme independently or who has to rely on other people.

4.9 Provide sample results in executable format

A sample result of each assessment is provided as reference. When students play the sample result, they can see how it works and what their goal is. It is much clearer to show the purpose of the question rather than to only describe it on paper. A picture is worth a thousand words.

All the sample results are compiled as executable code. Students can implement it in any way, but it has to meet all the functions required in the question.

4.10 Survey Results

A survey has been given to the students who have finished their CBC course. Generally, the feedback is positive. Most (80%) prefer to learn VB as their first computer language. They (90%) like the textbook with more examples and exercises. 40% want much more time to learn programming. No one wants to shorten it. They all (100%) like learning programming in the computer room rather than classroom, because they can test their programming immediately. They use different ways, which mostly help them to understand programming. 40% use pseudocode, 40% use a flowchart, 20 % use real language. Some of them (60%)prefer a group assignment. Others (40%) prefer an individual assignment. One comment said group assignment provided a better chance for discussion. Another comment said that it depends on the group. All of them (100%) want open-book practical test and they (100%) also want sample results in EXE so they can see what the programme looks like or should do.

5. CONCLUSION

To teach adult students to both read and write programmes, first and foremost we need to choose a proper computer language. Programming environment and textbook are equally as important. The way of teaching needs to meet the students' different learning styles and cultural backgrounds. We can do this by the what (e.g. VB, flowchart and sample results), the where (e.g. computer room), the when (e.g. more teaching time) and the how (e.g. VKA, group and practical assessment) to create a fresh teaching and learning environment. Feedback from students can help us to continue improving our teaching.

In a word, teaching programming is not just giving students a manual to operate; it is using more creative

and logical thinking; it is at both the invention and creative level. Therefore, it is necessary for us to open our minds, in order to improve the way we teach. Meanwhile, we must keep our eyes and ears turned to the changes in computer language.

REFERENCES

- Fincher, S. (1999) "What are we doing when we teach programming?" 29th ASEE/IEEE Frontiers in Education Conference
- Jenkins, T. (2001) "Teaching programming: A journey from teacher to motivator", <http://www.ics.ltsn.ac.uk/pub/conf2001/papers/Jenkins.htm>
- Knowlton, T. (2002) "Using Visual Basic to prepare learners for C++", Tech Trends - Industry Articles, Thomson course technology. http://www.course.com/techtrends/VB_041801.cfm
- Levy, S. (1995) "Computer language usage in CS 1: survey results", SIGCSE bulletin, Vol 27, number 3.
- Lusty, G. (1993) "Bicultural issues and teaching methodologies in New Zealand polytechnics", NZCSA conference.
- Microsoft Corporation (1998) "Microsoft Visual Basic advances RAD industry Leadership position", <http://www.microsoft.com/corpinfo/1998/Jan98/vb5mompr.htm>
- Nelson, M. & Rice, D. (2000) "Introduction to algorithms and problem solving", 30th ASEE/IEEE Frontiers in Education Conference
- Norvig, P. (2001) "Teach yourself programming in ten years", <http://www.norvig.com/21-days.html>
- O'Connor, J. (1999) The beginners guide to adult teaching and learning, Tairawhiti Polytechnic
- Schneider, D. (1999) Essentials of Visual Basic 6.0 programming, Prentice-Hall, Inc.
- Trandis, H. et al. (1988) "Cross cultural training across the individualism-collectivism divide", The international Journal of intercultural relations, Vol 2, chapter 7