A South African Case Study of University Computer Literacy Education[1]

Matthew C. Clarke
Department of Computer Science and Information Systems
University of Natal, Pietermaritzburg

Rosalie A. C. Wilson
Options Consulting
Pietermaritzburg

Noleen D. Loubser
Computer Assisted Teaching Centre
Pietermaritzburg

Abstract

Computer literacy education in South Africa faces numerous problems, among them the limited resources in comparison to the high demand and the diversity of student entry-knowledge of computers. Nevertheless there is a strong commitment to computer literacy within the University of Natal, and a broader national commitment to science and technology education. There is also a national commitment to restructuring the education process via the implementation of a National Qualifications Framework (NQF). This context prompted the Department of Computer Science and Information Systems at the University of Natal, Pietermaritzburg to rethink its academic computer literacy courses. This paper describes how the End User Computing (EUC) course was restructured in line with current education principles of criterion-referenced assessment, a learner-centred approach, outcomes-based learning and co-operative learning. The effectiveness of the restructured course is then discussed.

1. Introduction — The Context of Computer Literacy Education in South Africa

As part of its transformation into a modern democratic country, South Africa is paying a lot of attention to the revision of its educational policies. The government's various-coloured Papers on education echo the preceding report by the National Commission of Higher Education (NCHE) in their vision of a single co-ordinated system of higher education for South Africa, subjected to a far greater degree of government direction than is the case at present. This vision includes a planned National Qualifications Framework (NQF) which, when established, will require all educational qualifications to be registered through the South African Qualifications Authority [EDUC96]. Even universities will register their qualifications according to the number of credits that are required at each level of study, their educational outcomes, the methods whereby these outcomes are to be assessed and their articulation with other qualifications.

In addition to these dramatic changes in the South African education system, the teaching of most computer-related subjects are continually in need of modification due to the rapid changes in computing technology and the consequent integration of these technological changes into social patterns and structures. In this context it is not surprising that computer literacy education in South Africa requires rethinking.

Recent policy statements have committed the University of Natal to curriculum reform in line with the NCHE and NQF, and have consistently affirmed the need for all graduates to be familiar with information technology [VCR94, DUMI97]. This is a noble commitment, but one whose implementation faces numerous obstacles —

- Should computer literacy be seen as an academic course contributing credits towards a student's degree, or should it be seen as a required, but non-credit-bearing skill?

- The large class sizes which result from high student demand make it difficult to teach a topic with such a substantial practical orientation as computer literacy.

[1] This research was supported by the University of Natal's Research Fund.

- Changes in student demographics have lead to a situation in which students' prior level of exposure to electronic technology and their entry knowledge of computers are extremely diverse. It is difficult to structure a computer literacy course to suit a student body with such disparate entry skills.

- In the current climate of declining funds, any computer literacy course must make extremely efficient use of limited laboratory facilities.

2. Case Study — End User Computing at the University of Natal, Pietermaritzburg

The contribution of the Department of Computer Science and Information Systems to the University of Natal's commitment to computer literacy is primarily in the form of two credit-bearing courses — Introduction to Computing (a 2-credit course) and End User Computing (a 4-credit course). The latter, EUC, is intended primarily for students in the Faculty of Commerce (about 400 students annually) but is also open to other faculties (about 250 students). EUC may be the only computer course in the student's degree or may be the start of a major in Business Information Systems.

The Department has perceived various inadequacies of the EUC course for several years, partly with respect to its content but primarily with respect to its teaching approach, and in late 1995 agreed to restructure the course. The purpose of this restructuring was threefold — to improve the quality of the course without placing extra load on teaching staff; to increase the pass rate; and to test the appropriateness of various educational techniques within the constraints of time, academic expectations, student abilities and computing resources. During the first half of 1996, the first author co-ordinated this restructuring in collaboration with other members of the Department and various educational specialists. The course was implemented in the second half of 1996 and evaluated by an external consultant — the third author.

The new course was designed using the following broad strategy —

1. Understand the constraints within which the course must operate.
2. Define the course content via a clear statement of the aim of the course and a detailed list of objectives.
3. Choose teaching methods which are appropriate to the course objectives and which maximise the active participation of students.
4. Design assessment which tests whether the objectives have been achieved.
5. Evaluate the effectiveness of the course.

The following sections comment on each step in this strategy.

2.1. Constraints

Given the role of universities, as distinct from other tertiary education institutions, it is necessary that a credit-bearing course in computing have some academic substance. That is, it cannot be a merely skills-based course teaching students how to use a computer, but must inculcate some deeper understanding of how a computer works and of the role information technology plays in organisations and society.

Perhaps like most tertiary education institutions, there is a large demand for computer literacy courses in the University of Natal. In some highly developed countries, it may be possible to make computer literacy a pre-requisite to tertiary study, but in South Africa this is not yet common practice. We have little or no control over student numbers nor their entry knowledge and hence must expect large classes of students whose prior exposure to computers is very diverse.

Laboratoriess of PCs are available for the supervised practical sessions (33 students at a time) and for students' unsupervised work on practical assignments. These computer labs have placed a severe limitation on teaching for several years and have only permitted us to provide six supervised practical periods for each student through the course.

The University imposes some administrative and logistical constraints such as sessional dates, timetable periods and venues. EUC was historically allocated three 45-minute lecture periods for each of thirteen weeks, plus an extra weekly period for practical and tutorial work. There was also a legacy of software, lecture material, course notes and a prescribed textbook, but we did not consider these to be unchangeable.

2.2. Course Content

The Overall Aim of the EUC course was redefined as follows —

> The purpose of this course is to provide students with knowledge about principles of computer hardware and software, the use of common packages, and issues related to the use of computers as a personal productivity tool. The course is practical in nature and oriented towards using computers to solve problems which students are likely to encounter in their future places of work rather than the design or programming of computing systems.

This Aim is supported by a lengthy list of Intended Learning Outcomes which is split into three sections — outcomes related to practical skills in the use of computers, theoretical knowledge of computer and information systems concepts, and more general life-skills such as the ability to assimilate, organise and communicate knowledge.

We favoured the term "Intended Learning Outcomes" over the more traditional "Course Objectives" for several reasons. First, in accordance with the educational literature, "Objectives" must be stated behaviourally, whereas "Outcomes" can include cognitive and attitudinal aspects of the learning process. There are certainly behavioural outputs of EUC (such as the ability to use word processing software effectively) but it was also our intention for students to learn higher level critical thinking skills and inter-personal skills which cannot be expressed so easily in behavioural terms. Second, it is appropriate to emphasise that the outcomes are related to the individual student's learning rather than to the course. A "course" is not the sort of entity which can achieve any objectives; rather, a course is a structure which is designed to assist the learner to achieve certain ends. The student must take an active role as learner in order to achieve their desired outcomes, rather than passively watch as the course progresses towards its objectives. Third, these outcomes indicate the Department's *intentions* for what students may learn during the course. The Department can provide resources to help students fulfil these intentions, but it is not finally responsible for whether the intended outcomes are attained — that responsibility must rest with the individual student.

The central role of these Intended Learning Outcomes highlights our commitment to the outcomes-based approach required by the NQF. The detailed content of the course, its teaching methods and forms of assessment are all derived from the Intended Learning Outcomes. All class meetings (Lectures, Pracs and Tutorials) are explicitly presented as resources to enable students to achieve the advertised Intended Learning Outcomes.

As indicated in the previous section, an academic course such as EUC must aim for more than just skills training. For this reason, the Intended Learning Outcomes make no mention of any particular hardware or software. It is not the aim of the course to teach the syntax of any particular operating system or software package, but rather the underlying concepts of human-computer interaction techniques. The syntax will soon be outdated and of no use to a student who ends up in a job where different software is used, but students who have learnt the underlying semantics will be able to transfer that knowledge to other packages. Of course some *actual* user interface (or preferably several different user interfaces) provides the basic experience for the student, but the learning of that user interface is not the goal. The students' practical experience is used as the basis for learning about deeper semantic issues such as the process of selecting objects and applying actions to them, the need to organise files into directories or folders, the concept of selecting options from a menu, and the process of integrating information from different sources. The course explicitly compares the syntax of different user interfaces (e.g. deleting a file in both MS-DOS and Windows) in order to highlight the common semantics. Thus the Intended Learning Outcomes would remain unchanged if the course is taught one year with MS-DOS and Word Perfect and the next year with Windows-95 and MS Word.

The second area in which EUC delves more deeply than a skill-based computer literacy course is the theoretical understanding of *how* a computer stores and processes information. For instance, the course covers technical terminology related to data representation and storage, conversion between binary and decimal, the inter-relationship between various hardware factors with respect to how they affect processing speed, and the interpretation of data transmission statistics. These theoretical concepts are not taught in as much detail as one would expect in a computer science course, but their inclusion moves the student beyond questions such as "How can I get it to do what I

want?" and on to "How does the computer actually do this?"

Thirdly, EUC goes beyond a skill-based course in its coverage of organisational and social aspects of information technology. The course looks at computing systems rather than just computers, and promotes an understanding of the human element of such systems.

2.3. Teaching Methods

Teaching methods were chosen which were both appropriate to the Intended Learning Outcomes and which maximised active participation.

In line with the practical orientation of the course, we increased the time students spent under supervision in the computer laboratory fourfold, from one 45-minute period every second week to two 45-minute periods per week. If it had been logistically possible, we would have increased this time even more. These double-period Prac sessions are *taught* rather than simply *supervised*. The lesson plans are prepared by a senior staff member, but often taught by a junior member. The Prac sessions make use of an overhead projection of the teacher's computer screen and run in a demonstrate–copy cycle. When the Prac is particularly complex, notes are distributed to enable students to repeat the Prac in their own time.

In accordance with Kolb's experiential learning model [KOLB84, SVIN87], the Pracs provide the foundation of the course, rather than the lectures. The Pracs do not just train the students in practical skills but also provide the experiences on which the abstract theoretical concepts can be built.

Whereas the EUC course had previously taught a variety of powerful commercial software packages (such as Word Perfect, Quattro-Pro and dBXL), the Pracs in the new course are based on a single, integrated package (MS-Works). The advantages of such a package for a first course in computing are great. Most importantly, the package provides a consistent user interface for word processing, spreadsheets and database. This save an enormous amount of time and confusion since once the word processing component has been learnt, the students already know a large amount about how to use the other components. Another advantage is that MS-Works is not so feature-full as the software used previously, allowing the teaching to focus on the fundamental concepts and features rather than confusing the beginning student with a complex plethora of rarely-used menu options, icons, keyboard shortcuts, macros and settings dialogs.

We also decided to replace one of the three Lectures with a Tutorial. The Tutorial groups are small enough to allow greater student participation and promote a larger degree of both dialog with the teacher and peer-learning. The Tutorials groups are split further into small work groups of about four students. These work groups remain together throughout the course, working co-operatively on Tutorial worksheets and assignments. The worksheets typically require the groups to draw on information from the previous Lecture and from the textbook in order to apply the knowledge to some practical situation. For instance, in the section on data communications, they may be asked to calculate how many minutes it would take to transmit a full stiffy-disk of information from one computer to another over a modem operating at 9600 bps.

Although Lectures are less important in this course than in traditional university courses, they still serve an important role. Lectures are not used as the primary means of transmitting detailed information to students, but rather to paint broad pictures of the fundamental issues. A Lecture might describe the purpose of a modem and its role in one or two appropriate case studies, but the students must fill in the details of modulation, demodulation and transmission rates from the textbook in their own time. Lectures seem to us to be good for setting expectations and attitudes, indicating which issues are important and how one concept relates to another, and for motivating student interest through brief case studies, but not for the dictation of fine detail.

2.4. Assessment

Grading the course is based directly on the assessment of whether students have achieved the Intended Learning Outcomes. The marking is criterion-referenced rather than norm-referenced, and places a larger emphasis on continuous assessment rather than on a final exam. The overall grade is derived from the components shown in Table 1.

The six Real-Life Cases are assignments which are designed to be as realistic as possible for the business and organisational jobs in which graduates are likely to find themselves. For example, the two assignments completed in the small workgroups involve the analysis of a small business's need of computers and a

recommendation of what computer system they should purchase.

Table 1: Breakdown of EUC Assessment

20%	Four Real-Life Cases completed by individuals.
10%	Two Real-Life Cases completed by small groups.
10%	Four multiple choice quizzes.
20%	Two practical tests.
10%	A one-hour, open-book Written Test.
15%	A one-hour, computer-mediated, multiple choice exam on facts, terminology and concepts.
15%	A one-hour open-book written exam requiring the application of memorised and textbook knowledge to realistic situations.

Each group assignment receives a single mark, but at the end of the course the group completes an evaluation of each members' contribution. Each group member's personal mark is derived from the group mark, the peer-evaluation, and their frequency of attendance at Tutorials. This mark-distribution scheme is a modification of one suggested in [CONW93].

The final examination is split into two components — one which focuses on the memorisation of computer terminology and basic concepts, and the other which assesses the extent to which the students can apply what they have learnt to practical situations. The first follows the same style as the four quizzes and is administered via computer using software which allows a variety of multiple-choice and short answer questions. The second is an open-book exam in the same style as the earlier Written Test and the Tutorial worksheets. For a complete discussion of this style of examination, see [GRAY97].

3. Evaluation of the Restructured Course

The key innovations in the present EUC when compared to its precursors are listed in Figure 1. The intention of the evaluation process was to indicate the extent to which these innovations were effective in reaching the goals described at the beginning of Section 2 above.

- Detailed Intended Learning Outcomes.
- Increased supervised time in computer labs and reduction in lecture time.
- Demonstrate–copy approach to computer lab sessions.
- Use of a low-end integrated software package rather than separate high-end word processor, spreadsheet and database packages.
- Small group work.
- Changes to assessment —
  - Increased continuous assessment, including practical tests and small group assignments.
  - Decreased emphasis on final exam.
  - Open-book class test and final exam.
  - Computer-administered multiple choice quizzes and exam.

Figure 1. Summary of Innovations in EUC

An external computer training consultant conducted a participant observational study [GOOD52] to evaluate these innovations. The evaluator attended three Lectures, five Tutorials and seven Pracs incognito, giving her the ability to interact with these students and get feedback from them about the course in an informal way. Students also completed two evaluation questionnaires during the course, and the course was discussed with the EUC lecturers. The main conclusions from this external evaluation are as follows —

> The greatest perceived changes and benefits of the EUC course have been in the area of practical application. The increased laboratory times have allowed the students to confidently master the skills of basic computer literacy. At the same time, they have acquired a marketable skill for life after they leave the university. In contrast, the lectures have not been viewed as positively, and certain modifications will be necessary in future courses. There have been mixed feelings about the tutorials, and the expected uneasiness in reaction to collaborative learning — the group work.
>
> Most probably, the biggest problem has been in integrating the three components

> of the EUC course — there has been much difficulty in seeing how these go together. One possible cause has been the radical move from content to concept, not in itself an issue, but very likely too much, too fast, with an unrealistic avoidance of content altogether. I would recommend that a certain amount of basic content be permissible, and used as the structure on which to build conceptual ideas, especially as this is a first-year course, and many students come out of a content-driven school system. [LOUB97]

In terms of student satisfaction, the evaluation provided mixed results. There were some very positive attitudes towards the increased Prac time, and it seemed that the Intended Learning Outcomes made the aims of the course much clearer. But students also seemed more confused about what was required of them and consequently less satisfied with both the course administration and the course assessment. Students expressed surprise at the amount of work the course demanded, but we have always found that computing courses require more time than other subjects. It may well be the case that the changed teaching methods were too radical for the students and implemented too quickly.

In terms of staff satisfaction there is a continued commitment to the changes, but some disappointment that the pass rate for the course did not change significantly from the previous year. Although we kept detailed records of the hours spent on the course and its predecessor by lecturers, tutors and support staff, it is difficult to compare these figures. The class sizes were significantly different, and the time required to establish a new course is always greater than that required to repeat a well-established one. The changes certainly led to an increase in the time spent on administration (for instance marking and recording marks for continuous assessment was more time-consuming than for the previous course which relied more on the single final exam) and responding to student queries outside class time. On the other hand, it is expected that the detailed lesson plans which have now been written for all Lectures, Pracs and Tutorials will significantly reduce the preparation time in the future.

Conclusion

In line with current trends in South African education, the End User Computer course at the University of Natal, Pietermaritzburg was restructured using an outcomes-based approach. Once the Intended Learning Outcomes were defined, appropriate teaching and assessment methods were chosen and the course was implemented during 1996. This new course increased the Prac time by a factor of four, used a low-end integrated software package rather than separate high-end packages, dropped one of the three Lectures but added a Tutorial, introduced small group work, and placed more emphasis on continuous assessment.

The course was evaluated by an external consultant whose recommendations are being addressed at the moment. In order to improve the new course during its second running, we have presented the Intended Learning Outcomes to students gradually rather than overwhelming the students with the whole course plan at the outset. We have given the students much more direction about the textbook readings — by making explicit reference to the readings during Lectures, publishing a week-by-week reading list, and making a stronger connection between the textbook and the Tutorial worksheets. We have moved a discussion of group dynamics from the end of the course to an earlier position so that students can incorporate the ideas into their small workgroups. We have engaged senior students to act as mentors and are considering using the third Lecture period for remedial teaching.

It is hoped that these refinements will make the course even more effective and that the Intended Learning Outcomes may serve as a starting point for future NQF computer literacy course specifications.

References

[CONW93] Conway, R., Kember, D., Sivan, A. and Wu, M.; Peer Assessment of an Individual's Contribution to a Group Project; in *Assessment and Evaluation in Higher Education*, v18 #1, 1993.

[DUMI97] Duminy, A. and Lund, J.; Curriculum Reform; in *Update* Jan 1997, pub. University of Natal.

[EDUC96] Understanding the NQF: A guide to lifelong learning; Education Information Centre and the Independent Examinations Board, 1996.

[GOOD52] Goode, W. J. and Hatt, P. K.; Methods in Social Research; McGraw-Hill, 1952.

[GRAY97] Grayson, D. J. and Clarke, M. C.; Improving Examinations: A Cooperative Exercise Between Two Academics; In preparation.

[KOLB84] Kolb, D. A.; Experiential Learning: Experience as the source of learning and development; Prentice-Hall, 1984.

[LOUB97] Loubser, N. D.; Evaluation of End User Computing (EUC) Course; Unpublished report, 1997.

[SVIN87] Svinicki, M. D. and Dixon, N. M.; The Kolb Model Modified for Classroom Activities; in *College Teaching*, v35 #4, 1987.

[VCR94] The Vice-Chancellor's Review. Phase II: Planning Guidelines 1994-1998; University of Natal, 1994.