Proceedings

SACLA 2004

Sunday 4 July – Tuesday 6 July 2004

Rembrandt Klopper and Manoj Maharaj (Editors)
Table of Contents

Requirements for a Software Development Facility for a Research Centre: A Preliminary Investigation .................................................................................................................. 6
Harnessing GIS to extend HCI teaching .............................................................................................................. 16
Applicability of the Technology Acceptance Model in three developing countries: Saudi Arabia, Malaysia and South Africa ............................................................................ 33
e-Government: ICT Considerations for Citizen Service Delivery and Communication in a Metropolitan Municipality in South Africa ........................................................................... 43
The ECT act and it’s impact on a typical computing curriculum ......................................................................... 60
Knowledge Management – What Education Can Learn From Corporate Memory Systems ........................................... 72
Using Cluster Analysis to Investigate the Role of Culture in the Adoption of Web-based Learning Tools .............................................................................................................. 104
The Design, Implementation and Evaluation of a cost-effective Knowledge Management System – A South African Case Study ........................................................................ 118
The Application of the Cognitive Dimension Framework for Notations as an Instrument for the Usability Analysis of an Introductory Programming Tool .................................................................................................................. 134
Introducing BCom Information Systems Students to Basic Programming: Overcoming some of the Difficulties .................................................................................................................. 161
Friendly machines: interaction-oriented robots today and tomorrow .................................................................. 170
The South African Computer- and Information Science scholarly community: do we practice what we preach? ........................................................................................................................................ 189
Determining suitable programming language(s) for the B.Tech. (IT) degree .................................................. 204
e-Learning artifacts: are they based on learning theory? ...................................................................................... 224
A framework for determining Ethical Decision-Making for interaction at Tertiary Institutions .................................................. 245
Intelligent Agent Driven Bandwidth Management for Online Content Delivery in a Wireless Environment: A Conceptual Study for Distance Learners ........................................................................... 268
Impact of Personality Typing on Teaching Strategy in an Information Systems Major ........................................ 277
Internet Technology Paradigms in Tertiary Educations: Experiences with Web Based Learning in a Residential University Environment ........................................................................ 292
Incorporating Industry Curricula into Tertiary Education .................................................................................. 306
Student perceptions of learning Object-Oriented Programming ........................................................................ 324
An Exploratory Study of Online Shopping Behaviour of South African Consumers ........................................ 336
The Development and Evaluation of an Iconic Programming Tool for an Introductory Programming Course .................................................................................................................. 352
Affective Aspects in Learning and Teaching Programming .............................................................................. 366
An Investigation into the Thinking Styles of Information Systems and Technology Students ............................................. 379
The South African Computer- and Information Science scholarly community: do we practice what we preach? ........................................................................................................................................ 395
Evaluating an e-Learning IT Project Management Course: A Social Constructivist Perspective .................................................. 408
Future Communications ................................................................................................................................... 425
Visual aesthetics and its effect on communication intent: a theoretical study and website evaluation ................... 451
The Determination of Successful Teaching Programs: A Study of Irish and South African Part-Time Undergraduate Information Systems Courses ........................................................................... 470
**List of First Authors**

<table>
<thead>
<tr>
<th>Name</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adigun, MO</td>
<td>6</td>
</tr>
<tr>
<td>PATRICIA MARGARET ALEXANDER</td>
<td>16</td>
</tr>
<tr>
<td>Udo Richard Averweg</td>
<td>33</td>
</tr>
<tr>
<td>Udo Averweg</td>
<td>43</td>
</tr>
<tr>
<td>Andries Barnard</td>
<td>60</td>
</tr>
<tr>
<td>S. Berman</td>
<td>72</td>
</tr>
<tr>
<td>Irwin Brown</td>
<td>104</td>
</tr>
<tr>
<td>André Calitz</td>
<td>118</td>
</tr>
<tr>
<td>Charmain Cilliers</td>
<td>134</td>
</tr>
<tr>
<td>Lyrice Cohen</td>
<td>161</td>
</tr>
<tr>
<td>Stephen J. Cowley</td>
<td>170</td>
</tr>
<tr>
<td>Ms. Jennifer De Beer</td>
<td>189</td>
</tr>
<tr>
<td>Mr. JO Dehinbo</td>
<td>204</td>
</tr>
<tr>
<td>M R de Villiers</td>
<td>224</td>
</tr>
<tr>
<td>S Dimas</td>
<td>245</td>
</tr>
<tr>
<td>Mike Eccles</td>
<td>277</td>
</tr>
<tr>
<td>Mike Eccles</td>
<td>292</td>
</tr>
<tr>
<td>Peter Gallert</td>
<td>306</td>
</tr>
<tr>
<td>Leila Goosen</td>
<td>324</td>
</tr>
<tr>
<td>Miss A Gordon</td>
<td>336</td>
</tr>
<tr>
<td>Jéan Greyling</td>
<td>352</td>
</tr>
<tr>
<td>Ken Halland</td>
<td>366</td>
</tr>
<tr>
<td>Rikesh Harypursat</td>
<td>379</td>
</tr>
<tr>
<td>Jennifer De Beer</td>
<td>395</td>
</tr>
<tr>
<td>Pieter Joubert</td>
<td>408</td>
</tr>
<tr>
<td>Kirstin Krauss</td>
<td>451</td>
</tr>
<tr>
<td>Mr. JC Lannon</td>
<td>470</td>
</tr>
<tr>
<td>Rut Laubscher</td>
<td>485</td>
</tr>
<tr>
<td>Hugo Lotriet</td>
<td>506</td>
</tr>
<tr>
<td>E. Venter</td>
<td>526</td>
</tr>
<tr>
<td>Manoj Maharaj</td>
<td>543</td>
</tr>
<tr>
<td>B.J. Mallinson</td>
<td>550</td>
</tr>
<tr>
<td>Linda Marshall</td>
<td>566</td>
</tr>
<tr>
<td>Mr. HD Masethe</td>
<td>585</td>
</tr>
<tr>
<td>Muzi Matyila</td>
<td>595</td>
</tr>
<tr>
<td>June D. Mercer-Chalmers</td>
<td>606</td>
</tr>
<tr>
<td>Peter L Mkhize</td>
<td>631</td>
</tr>
<tr>
<td>Mokwena SN</td>
<td>643</td>
</tr>
<tr>
<td>Deshendran Moodley</td>
<td>654</td>
</tr>
<tr>
<td>K.A. Naudé</td>
<td>670</td>
</tr>
<tr>
<td>E J Naudé</td>
<td>688</td>
</tr>
<tr>
<td>Shaun Pather</td>
<td>698</td>
</tr>
<tr>
<td>S.R. Ponelis</td>
<td>710</td>
</tr>
<tr>
<td>Laurette Pretorius</td>
<td>731</td>
</tr>
<tr>
<td>Tanja Reinhardt</td>
<td>748</td>
</tr>
<tr>
<td>Nelishia Pillay</td>
<td>748</td>
</tr>
<tr>
<td>Schoeman I.L</td>
<td>773</td>
</tr>
</tbody>
</table>
Ravina Sewlal .............................................................................................................................. 793
Lisa Seymour ............................................................................................................................ 802
UG Singh .................................................................................................................................. 820
Mr. Shawren Singh ................................................................................................................ 832
Jean-Paul Van Belle ................................................................................................................. 875
Alta van der Merwe ............................................................................................................... 895
Dr. Johan van Loggerenberg .................................................................................................... 918
T.M. van der Merwe ............................................................................................................... 930
Managing Programming Assignments in the Computer Science Classroom

Schoeman I.L., Department of Computer Science, University of Pretoria, lschoeman@cs.up.ac.za;
Pieterse V., Department of Computer Science, University of Pretoria, vpieterse@cs.up.ac.za;

Abstract

In this paper we investigate and analyze opinions and strategies concerning the role of practical assignments in programming courses. While the general opinion that practical assignments form an important part of any programming course cannot be disputed, various views on the purpose of these assignments, as well as the manner in which it should be implemented, exist. Data have been gathered by means of two successive online questionnaires sent to SACLAM members. A qualitative questionnaire requesting opinions on specific aspects of practical assignments was followed by one where responses to the first questionnaire were quantified and the respondents were asked to rate the opinions. Opinions could eventually be grouped into three large areas, the first being the purpose of these assignments where the general consensus was that it constituted a vehicle for applying concepts introduced in class. Many views on the problem of plagiarism were voiced and remedies suggested, which emphasized the impact of such a destructive practice. To conclude, opinions on group work were generally positive although some problems were identified.

Introduction

Practical assignments form an integral part of any programming course. As any lecturer realizes, it is of the utmost importance that students acquire practical experience in the implementation of the programming principles they have studied. Acquiring programming skills, albeit an art or a skill, necessitates many hours of concentration, perspiration and frustration. Often students revert to the less taxing skill of simply copying the work of others, or in the case of group work, not contributing significantly.

Background

Although a general consensus exists that practical assignments should form part of any programming course, the purpose of these assignments, as well as the way in which it is administered, differs significantly. Assignments can be completed individually, in pairs or in groups.
It can be seen as hands-on training, the practical implementation of theoretical concepts or as a way of assessment. Students often do not realize the value of practical work in the process of acquiring problem-solving skills in programming. A quest for marks often replaces the quest for knowledge. Where this mindset exists, the problem of plagiarism repeatedly rears its ugly head, thus defeating the entire purpose of practical assignments. Much effort on the side of lecturers is directed towards eradicating this undesirable practice, and a large variety of opinions exist in this regard.

In some courses practical work take the form of group projects, in which case a whole new set of rules apply. It is generally agreed that group work is necessary in order to develop skills to deal with this kind of setup as it is experienced in the workplace. However, new circumstances bring about new problems and again a large variety of opinions of how to deal with situations that can arise, is voiced.

We therefore came to the conclusion that a survey among lecturers of programming courses where they could voice their own opinions and assess the opinions of others, would prove an interesting and enlightening exercise.

Aim

In this paper we investigate and analyse opinions on practical work and strategies used and suggested to deal with problems. Problems associated with the use of practical assignments are identified, and suggestions of how to deal with the problems in question, are discussed.

Method

We combine quantitative and qualitative research. With quantitative research data are gathered mostly by structured questionnaires. The responses are analysed using statistical measures. To ensure reliable results considerable effort has to be devoted to the content of the questions. With qualitative research data are gathered through interviews, observation, open ended questions, etc. The responses are analysed and assessed in greater detail. Schulze (2003) argues that educational researchers require both modes of inquiry to advance their understanding of teaching, learning and other human phenomena. The strengths of both these methodologies are harnessed by combining them in our research endeavour.

Data were gathered in two rounds. For the first round we invited all members on the e-mail list of SACLA to participate in an on-line questionnaire containing open-ended questions. We compiled a questionnaire using our own experience as well as insights obtained through observation and some informal conversation with close colleagues. The scope of the questionnaire was kept small in an
attempt to maximise participation, the main focus of this round being to gather qualitative data. Questions comprising the second round were based on data gathered during the first round.

For the second round the invitation to participate was sent to the same people. The questionnaire for this round contained a number of paraphrased statements made by the participants of the first round. Participants of the second round had to rate each statement according to its importance in their own opinion. The intention of this round was to establish quantitative agreement regarding the statements that were aired during the first round. Data gathered in this round was used to substantiate the statements in our conclusion.

Validity, Reliability and Ethical Measures

All participation was voluntary and anonymous. Moreover, in order to eliminate bias in the results, responses to the open ended items were analysed as academic peers, thus ensuring reliability.

Sample

The sampling method we used for our research resembles convenience purposeful sampling as described in the methodology manual published by the State of Texas, available online at http://www.sao.state.tx.us/Resources/Manuals/Method/download.html.

The character of the sample can be seen as purposeful since the invitation was sent to a targeted audience on the SACLA mailing list. Members of the list are presumed to be scholars interested in lecturing computer science and related subjects, hence being experienced in lecturing courses involving programming. The convenience character of the sample is attributed to the fact that the survey incorporated only lecturers on this mailing list, which means that it may not represent all lecturers, and may exclude knowledgeable scholars.

The mailing list has 222 members of which only 17 (7.66%) answered the first questionnaire, and 30 (13.5%) participated in the second round.

Identification of Problem Areas

The following questions were asked in the first questionnaire:

1. What do you see as the purpose of practical assignments?
2. Do you feel the above aims are met?
3. What do you see as the biggest problem that defeats the purpose of practical assignments?
4. What percentage of the final mark do you think should be allocated practical work?

5. What strategies do you employ to ensure that students do their own work?

6. What is your opinion / experience on group work for larger assignments?

7. What strategies do you employ to ensure that each member of a group is evaluated according to his contribution?

8. Does your department have a policy on copying / plagiarizing. If so, describe it briefly.

The answers to these questions were analysed, evaluating ideas that were aired by the participants and incorporating the outcomes in the second questionnaire. Participants were then asked to rate the statements in an attempt to quantify the findings. The following summarises the ideas that crystallised from the research.

The purpose of practical assignments

Participants formulated the purposes served by practical assignments designed and assessed with the set goals in mind, as follows. Statements are sorted from most important to least important according to the ratings assigned to them by the respondents.

a. Concepts explained in class are consolidated and related issues are explored.
b. The students' programming skills are developed.
c. Students are given the opportunity to have fun and acquire a sense of achievement.
d. A professional approach to coding is instilled.
e. The students' levels of cognition of the concepts are measured.

It is interesting to observe that most of the lecturers see practical assignments not primarily as an assessment instrument, but, in the true academic sense, serving a much more elevated goal.

The problems we face

The problems that defeat the purpose of practical assignments are closely related to the goals we would like to achieve. The following problems were mentioned and are listed in the order of importance as indicated by the respondents

a. Copying of code.
b. The absence of a learning culture. (student attitude and immaturity)
c. Inadequate policing and techniques to detect plagiarism.
d. Lack of capacity (limited laboratory space, not enough personnel, etc.)

We expected a higher rating for problems that might hinder the achievement of the goals listed in the previous section. The lack of capacity and absence of a learning culture can have a huge influence on the realization of all the goals, except for the use of practical assignments purely for assessment. Rating the lack of capacity lower, can possibly be attributed to the fact that most of the respondents have adequate capacity.

Assigning the highest rating to copying, illustrates the frustration and feeling of helplessness most lecturers experience in the face of the blatant disregard of many of the principles supposed to underlie the academic ideal. The copying of code can be seen as a symptom of the absence of a learning culture, constituting a hurdle obscuring the realisation of all the goals.

**Measures to Counter Plagiarism**

Plagiarism is a major impediment to the realization of our goals. Respondents proposed various strategies to address this problem. Their approach ranged from philosophical educational to practical or technical. Figure 1 shows the rating respondents assigned to the different proposed strategies. There are clear tendencies in these ratings which is summarised in table 1.

![Figure 1: Rating of Measures to Counter Plagiarism](image)

<table>
<thead>
<tr>
<th>Measure to Counter Plagiarism</th>
<th>Tendency of Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enforce a strict policy</td>
<td>High</td>
</tr>
<tr>
<td>Plagiarism Detection</td>
<td>High</td>
</tr>
<tr>
<td>Allocating Minimised Weight</td>
<td>Undecided</td>
</tr>
<tr>
<td>Action</td>
<td>Category</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Enforcing a strict anti-plagiarism policy.</td>
<td>Punish</td>
</tr>
<tr>
<td>Computer based plagiarism detection.</td>
<td></td>
</tr>
<tr>
<td>Minimized weight allocated to practical assignments</td>
<td>Avoid</td>
</tr>
<tr>
<td>Evaluation by means of practical tests</td>
<td></td>
</tr>
<tr>
<td>Not providing model answers to be used as templates for practical assignments.</td>
<td>Prevent</td>
</tr>
<tr>
<td>Random problem generation.</td>
<td></td>
</tr>
<tr>
<td>Include a declaration of authenticity to be signed by the students.</td>
<td>Delegate</td>
</tr>
<tr>
<td>Motivate students to work individually by explaining the benefits of learning by doing.</td>
<td>Educate</td>
</tr>
<tr>
<td>Support to improve students’ confidence in their ability to complete practical assignments successfully</td>
<td></td>
</tr>
</tbody>
</table>

We sorted these measures in the categories shown in table 2.

Table 2: Categories of Measures to Counter Plagiarism

Punish

This solution proposes that measures should be taken to identify the offenders and punish them harshly. Plagiarism detection has received much attention and many detection programs are available. One example is the Online Judge created by Cheang et al (2003) which is able to
effectively detect cases where students plagiarized by adding redundant code, changed variable/method names or swapped the order of statements. We agree that such measures will discourage most of the students to plagiarise. It should however be noted that the implementation of such measures will also stimulate the dishonest student to direct considerable effort towards inventing ingenious ways to avoid detection.

Avoid

The philosophy behind these strategies is to minimise the marks earned for practical exercises as a percentage of the total marks. The urge to copy should consequently diminish. Almost all the responses to which weight should be allocated to practical work, indicated a weight of 40% or lower.

Subsequent evaluation to determine whether students have mastered the concepts and acquired the underlying skills, will ensure that students complete the assignment. The downside of this approach is that can be difficult to include measurement of all the desired outcomes in the test. Moskal (2000) further points out that while formative assessment and the assignment of a numerical weight (e.g. a test or exam score) can be used to determine the degree to which criteria have been met, this method does not give students any indication of how their performance can be improved.

The average student attitude to strive towards maximum marks with minimum effort may lead to a tendency not to put enough effort in practical exercises which are only assessed by means of an after-test. They will rather revert to memorising solutions created by fellow-students. While avoiding the negative impact of plagiarism, the risk of neglecting the positive impact of students actively doing the assignments, is increased.

Prevent

In this category measures that will minimise the opportunities to copy and hence remove the temptation, are implemented. However, the individualisation of assignments through random generation may result in assignments of varying quality. Another proposed solution, namely withholding model answers, may deprive students of valuable study material.

Delegate

The responsibility of successfully completing the course, is delegated to the students. This is a legally sound solution, but cannot solve the problem on its own. Theoretically students are responsible for their own achievement. In the end, however, the lecturer stays responsible for the academic standard of the course and the ability of the students who have passed.

Educate

Educating students to apply ethical principles to the manner in which they approach their studies, is tightly coupled with the absence of a learning culture that was identified as a major impediment to the purpose of practical assignments. Unfortunately large classes and tight schedules prevent
lecturers from fulfilling their educational responsibility to create a learning culture and instil ethical values. Kourie (2004) emphasizes that curricula should aim to transmit an agreed-upon value-system, using skills training as a context to achieve this. Lecturers should constantly reinforce values relating to professionalism, responsibility, ethics, etc. This can be done by maximising opportunities to assimilate these values.

**Group work**

Group work is considered to comprise an essential part of any programming curriculum. In industry a group of programmers working on a project, constitutes the rule rather than the exception. It is therefore an important exercise to simulate this model in programming courses. Williams and Kessler (2000) experimented with the “pair-programming” model used in Extreme Programming (XP), an emerging software development methodology. They found that considerable benefit was derived from this model, among others that students learnt from one another, that grading was reduced and that there were less cheating, as it was now legitimate to copy code. The quality of the work was also better, as student pairs apply a positive form of pressure on each other thus improving student success and morale.

Responses in our survey were much along the same lines. The following benefits of group work in general were identified:

a. Working in groups leads to the enhancement of software quality
b. Large scale projects are impossible without group work

Where group work is specifically included in a programming curriculum, additional benefits were mentioned:

a. It contributes to the development of conflict resolution skills.
b. It reduces the assessment work load

![Figure 2: Rating of Benefits of Group Work](image)
As can be seen in figure 2, the benefit it holds for the students (developing their skills) is generally rated higher than the benefits it holds for the lecturers. Scott and van der Merwe (2003) mentions three "soft" IT issues that students are exposed to when working in teams, namely motivation, ethics and conflict resolution.

While group work creates opportunities for students the exercise these unique skills needed for the workplace, it can lead to a number of misuses like the following that was mentioned in the responses we gathered:

a. Group work exacerbates freeloading
b. Isolation of members can happen due to indifference or perceived incompetence.

Assessment of group work

Scott and van der Merwe (2003) lists the following aspects of group projects that needs to be assessed.

a. the quality of the product
b. the understanding of the underlying business problem and principles,
c. technical skills
d. soft skills
e. contribution of individual members to the group project.

In our study we only considered the latter. Special techniques are required to ensure that each member of a group is evaluated according to his contribution. When respondents were asked to elaborate on the strategies they employ to achieve this, the following two methods were the most prominent:

a. Peer evaluation: Members of the group assess one another.
b. Individual assessment: Individual students have to explain code or do some "real time programming" to demonstrate their skills, or have to write a test on the content.

The following were also mentioned:

a. Subdivide the assignment in separate deliverables that can be handed in individually.
b. Intra-Group evaluation
c. Enforce pair programming

For peer evaluation the most appropriate instruments seems to be scoring rubrics. Rubrics are described by Metler (2001) as rating scales or scoring guides with pre-determined performance criteria. When published in advance, it ensures that students are aware of the assessment criteria and associated standards, as well as encourages groups to do self-assessment. (Scott and van der Merwe, 2003)
Conclusion

The purpose of this survey was to investigate and analyse opinions on practical work and identify problems, useful strategies and suggestions. A second survey yielded quantitative results where the results from the previous survey were rated according to perceived importance.

In general the responses received showed true dedication and adherence to professional and ethical principles on the side of the lecturers. It was clear that the respondents were serious about their quest for quality in education. In keeping with the above, most respondents saw the purpose of practical assignments as the consolidation and practicing of programming concepts that were introduced in class and not primarily as an assessment tool.

Most respondents considered plagiarism and the copying of code to be major problems. The quest for quality education is again reflected in these views, as plagiarism defeats the entire purpose of giving practical assignments. Measures to counter the problem were quite diverse. We concluded that the ideal would be a learning culture where no one would even consider such a deplorable practice, but, as this is not feasible, several measures could be put in place to prevent plagiarism. There should, however, be a continual striving towards producing dedicated students with sound ethical values.

Group work was deemed to be important by most respondents. Again the benefits to the students were rated higher than the benefits it holds for the lecturers. The development of skills needed for a career in industry was also emphasized.

To conclude, this exercise proved valuable by providing fresh insights into the management of practical assignments as well as reassurance of the general state of our educational vocation.

References

http://www2.umassd.edu/SWPI/xp/pairprogramming/csed.pdf