ABSTRACT

Informatics 154 (INF154) is an introductory programming course at the University of Pretoria. The main objective of the course is to ensure that our students understand the fundamentals of program design and implementation. Administration poses a big challenge, especially because telematic and contact students need to be handled simultaneously. The diversity pertains to the fact that our students are enrolled for different degrees. The diversity in programming background requires a careful selection of the study material and assessment criteria for the course. The diversity also involves conducting four parallel streams of lectures in two languages, as well as making study material available to all students via the web. This research indicates that the overall goal of the course is met, but that adaptations should be considered regarding the student profile and the prescribed textbook.

1. INTRODUCTION

Informatics 154 (INF154) is an introductory programming course at the University of Pretoria. The main objective of the course is to ensure that our students understand the fundamentals of program design and implementation. Administration poses a big challenge, especially because telematic and contact students need to be handled simultaneously. The diversity pertains to the fact that our students are enrolled for different degrees. The diversity in programming background requires a careful selection of the study material and assessment criteria for the course. The diversity also involves conducting four parallel streams of lectures in two languages, as well as making study material available to all students via the web. This research attempts to assess the effectiveness of our efforts, i.e. whether our administrative procedures, teaching methods, the programming language we have chosen and the study material we use are enabling us to achieve our goal. The research data were collected by disseminating a questionnaire to all our students. We would like to acknowledge the inputs of Dr E M Louw and Statomet at the University of Pretoria with regard to the statistical analyses.

The INF154 course runs over a whole semester with a weekly one-hour lecture (for contact students) and a weekly two-hour practical session for both contact and telematic students. In order to meet the challenges of outcomes based education (OBE), specific outcomes are stated in the study guide, and continuous assessment is done using weekly class assignments and practical exercises. Because of its simplicity the Natural computer language is used for the practical work. In 2003 there were ca. 400 students enrolled for the course, 208 of whom completed the questionnaire, which was used to collect data for this paper.

2. STUDENT PROFILE

The course caters for a wide variety of students. Students who take INF154 are enrolled for various degrees. Most of the students study towards either the B.Com. (Informatics) degree (50.5%) or towards the B.Sc. (IT) degree (26.9%). Other smaller groups are: B.IS (6.7%), other B.Com. (5.8%), other B.Sc. (3.9%), B.IT (1%), BA (Information Science) (1%), other BA (1%), D.IT (0.5%) and other (1.9%). (Missing percentages are due to errors on the computer mark sheets used to answer the questionnaire). These data support the main focus of the course, i.e. programming for information systems. However, it was quite surprising that such a large percentage of B.Sc. students still take the course, although it is not a requirement. In a section on the questionnaire for “other comments” a few students indicated that the course also helped them to come to grips with programming concepts that are used in other programming courses. The fact that almost half of our students are not studying Informatics indicates that the scope should stay wide enough to make it relevant for these students.

Although INF154 is a first year course only 67.3% of the students are in their first academic year and only 58.7% are in their first historical year. While it may be a good idea for bona fide first year students to get marks for class assignments or class attendance, this may be frustrating for the senior students, and we should reevaluate this policy.
At present the course caters for both full-time and telematic students. However, only 3.4% of the students who filled in the questionnaire are telematic. According to our own records about 10% of the students do the course telematically, but the University’s system does not differentiate between the two groups, making it impossible to get precise statistics. The telematic study is facilitated by means of WebCT, which is also accessible for full-time students. This is such a successful platform for the distribution of class notes, assignments and notices that it will still be used, even when the telematic option is phased out in 2004.

Another aspect of diversity pertains to the language of preference. Historically, UP has been an Afrikaans university, and therefore it was quite surprising that only 24.5% of the respondents indicated that Afrikaans is their language of preference for the course. Up to now we assumed that the ratio was more or less 50/50. This suggests that we should print more English study guides and have more English lectures in a ratio of 3:1. However, it is interesting that the majority (61.4%) of those students who prefer Afrikaans as medium of tuition would like the practical assignments to be available in Afrikaans as well (at the moment all the practical material is provided only in English).

Our students’ academic backgrounds also differ a lot. Almost half of them (49%) felt that they already had an average to excellent level of programming expertise before starting the course, while the other half (51%) ranked their level of expertise as poor or non-existent. We try to accommodate students who can already program by providing a class exemption test at the beginning of the year, which was written in 2003 by 35 students, 27 of whom passed. Those who passed only had to complete a small project in Natural to obtain a semester mark and to write the final examination. The fact that half of the students indicated that their programming expertise is average to good, but that only 27 students qualified for class exemption, implies that there are still a lot of students in the theory classes and in the practical sessions who can already program. This is probably due to the fact that all students have to learn the syntax of Natural, but it poses a huge challenge to the lecturers to explain elementary concepts to students with poor programming background without boring the others to death! One possibility to address this challenge would be to divide the theory and practical groups according to level of expertise. Another possibility is to have extra tutorials for those students who need it.

The exemption test should stay an option, not only for students who passed Computer Studies (HG) at school, or another formal programming course, but also for students only with practical experience, because the survey showed that only 27.4% of our students did Computer Studies HG at school, yet 49% felt that they had an average to excellent level of competency in programming.

Despite all our efforts to cater for all the diversities in our student body, 37% still disagree that students’ different academic and cultural backgrounds are taken into consideration by the lecturers, while 26.4% are indifferent and 34.6% agree. The good news is that 74.5% understand why programming is relevant to their degree programs.

3. ADMINISTRATIVE CHALLENGES

One of the biggest problems with a large group of students is efficient course administration. It is not good enough to make announcements and arrangements during lectures because class attendance is not obligatory and declines progressively as the academic pressure builds up towards the end of the semester. The study showed that almost 70% (69.7%) of the students know which learning opportunities, such as preparation lectures, practicals and the tutor’s consulting hours, take place when and where. This is probably due to the printed study guides that are handed out at the beginning of the semester and due to the use of WebCT. WebCT’s discussion facility is used as the main portal for administrative notices.

All the class notes, assignments, memoranda, etc. are also published on WebCT. 76 %of the students responded that they knew where to find the study material. Even more, 80.3% said that they use WebCT daily or weekly to stay informed about the course and to get class notes, etc. This facility also saves the Department a lot of money in terms of printing costs. More evidence that WebCT is indeed used a lot, is the fact that 59.6% of the students responded that they are aware of the fact that the provided Natural software may only be used for educational purposes and not for business purposes, implying that almost 60% of the students read - and remembers! - the notice that was put on the discussion board at the beginning of the semester.

To provide 400 students with the software package that they use for practical programming is another administrative challenge. Besides being available in the laboratories, we have permission to distribute copies of the Natural software to our students for educational purposes. Those students who wanted a copy, had to order
one and pay a R20 handling fee, including the cost of the CD-R and of burning it. Quite a number of CDs were
defective and had to be returned and swapped. 57.7% of the respondents were satisfied with the way in which
the software was provided, while 18.8% were indifferent and 21.2% were dissatisfied.

Creating enough opportunities for communication between students and lecturers pose another big challenge for
such a course. We had three full-time lecturers, four assistant-lecturers and one tutor assigned to the course, and
between us we had 33 consulting hours per week, some of which were scheduled in the same time slots so that
there were at least one person available 21 hours per week. 52.9% of the respondents felt that there were enough
opportunities for communication, 22.1% were indifferent, and 23.1% felt that there were not enough
opportunities.

4. COURSE CONTENTS

By now the diversity that exists in the INF154 student body should be quite evident. Our hypothesis is that there
could be a difference in the perceptions of the two main degree groups who are enrolled for the course, B.Com.
versus B.Sc., regarding the contents, study material, teaching and assessment. The students' previous level of
programming expertise could also influence these perceptions. Therefore, we will focus on these aspects in the
rest of the paper. Unveiling these perceptions is of great value to the lecturers of INF154. It is they that have to
work with the students and manage the relationship between the teaching goal and teaching contents. It is also
important for them to unlock the content in a meaningful way to the student. There are especially three aspects
that they should account for, the reduction of the contents, the problem statement that will guide the learning and
the arrangement of the contents. The INF154 lecturers realised that these aspects could only be fulfilled when the
variety in the INF154 group is recognized, exposed and managed in order for the successful implementation of
the course.

A variety of factors were considered when selecting the programming language. A language with a simple
syntax and affordable license, especially for large groups, was the main selection criteria. The results of the
questionnaire regarding the intelligibility of the selected programming language (Natural) are very interesting.
The B.Com. group as well as the B.Sc. group experienced Natural as an easy language, as 61.1% of the st udents
in these two groups indicated that they agree with the statement that Natural is an easy language to understand.

By using the chi-square test (p-value of 0.0005) in comparing the opinion of B.Com. students with the opinion of
B.Sc. students on the intelligibility of Natural, it became evident that there is a highly significant difference
between these two degree groups at the 5% level of significance. In the B.Com group there was a much higher
degree of opposition the statement (see Figure 1). The reason may be that the B.Com. students lack a
programming background and therefore it could be more difficult for them to learn programming in general.

![Intelligibility of Natural by degree program](image)

Figure 1: The intelligibility of Natural according to students' degree program
If one divides the study regarding the intelligibility of Natural according to previous programming expertise an interesting picture appears. By using the chi-square test (p-value of 0.0006) in comparing the opinion of students having a poor (including very poor and average) level of prior programming expertise with those having good or excellent expertise on the intelligibility of Natural, it became evident that there is a highly significant difference between these two groups at the 5% level of significance. In the good expertise group there was a much higher degree of agreement with the statement that Natural is simple enough to understand easily (see Figure 2). This indicates that students who know other programming languages agree that Natural is simple and intelligible. It should also be remembered that the students with a poor expertise probably do not have knowledge of other programming languages to compare Natural with.

![Intelligibility of Natural by prior programming expertise](image)

Figure 2: The intelligibility of Natural according to students' level of prior programming expertise

From these results one can conclude that students who know other programming languages, either because they are studying B.Sc. (IT) or because they have prior programming expertise, would agree that the Natural language is indeed a programming language that is suitable to be used in teaching the basic programming principles to first year students.

When the students were confronted with the question whether they would prefer another programming language, such as Visual Basic, Java or C++, the following results were obtained. There was not a significant difference between the B.Com. and B.Sc. groups that would prefer another language and those who would not: 42.1% of B.Com. students would prefer another language, while 57.9% would not (or were indifferent), and 37.5% of B.Sc. students would prefer it, while 62.5% would not (or were indifferent).

When the language preference was tested using the expertise level of the students, the suitability of Natural for our purposes was emphasised. By using the chi-square test (p-value of 0.0001) in comparing the opinion of students having a poor (including very poor and average) level of prior programming expertise with those having good or excellent expertise regarding their preference for another programming language, it became evident that there is a highly significant difference between these two groups at the 5% level of significance. In the poor expertise group the majority of students disagreed to the statement “I would rather prefer another programming language such as C++, Java, VB, etc.” (implying that they do not want another language), but in the good expertise group the majority agreed (see Figure 3). The final conclusion is that students with no, poor or average computer background would not prefer another programming language as the teaching language. This indicates that the Natural programming language fulfils the objectives that this course sets out to fulfil (although the opposite is true for students with good or excellent prior programming expertise where the majority would prefer another language). Our main focus is to teach basic programming principles, concentrating on students with no, poor or average experience.
5. STUDY MATERIAL

A major problem though, for teaching purposes, is that since Natural is a proprietary language, there are no textbooks written on the language. In order to meet the challenge of providing suitable study material to the students, the textbook *Simple Program Design* (3rd edition) by L.A. Robertson was selected. The textbook was chosen on the grounds that it is language independent and uses pseudocode to discuss the basic programming concepts. The lecturers regarded the book as useful and applicable because it covers the whole syllabus and the Natural syntax is very similar to the pseudocode used in the book. However, although 60.1% of the students bought the textbook, only 6.7% often used the textbook, 29.8% used it occasionally, while 59.1% never used it. Further research explained the reason why. For the INF154 group in total 60.1% students indicated that the textbook was not useful to them (16.8% were indifferent). The trend was at first alarming because, although the lectures were structured to include work from the textbook, the majority felt that the textbook was of no or little help. However, from the “other comments” section, it became clear that some students felt that the class notes were sufficient and that the students would rather want a book on the Natural language itself. One can conclude that the nature of the textbook using no specific language contributed to the opposition of using the book. This conclusion is supported by the results of another question in which the students were asked whether the study material (textbook and course notes) were adequate to master the course content. Although the textbook was used seldom, 51.9% indicated that the study material was adequate, 20.2% were indifferent and 26% thought it was not adequate. This indicates that, for the group as a whole, the class notes are sufficient for the students’ purposes. However, more of the students lacking programming expertise (51.4%) than those having good or excellent expertise (35.7%) indicated that the study material was not adequate. Because there are no textbooks on Natural available, this gap may be filled by expanding the class notes and organising extra tutorials for those students who need it.

6. TEACHING AND ASSESSMENT

An analysis was done to compare the opinions of different categories of students on the weekly theory lectures (also called preparation lectures or exercise classes), but there was not a significant variance between the categories. The students were asked whether they found the theory lectures useful. Overall, regardless of category and ignoring faulty data, 68.7% indicated that these lectures were informative and necessary. This is contrary to our initial perception that probably two-thirds or more of students could cope with the study material by simply accessing and reading the lecture notes that are placed on WebCT.

The students are given one project, which they are expected to work on during the holidays. The purpose is to get them to practice what they have learnt in the first term of the course. The majority of both the B.Com. and the B.Sc. students indicated that the project helped them to better understand what had been covered in the preparation lectures and practical sessions. It is interesting that the B.Sc. students found it even more useful than
the B.Com. students. By using the chi-square test (p-value of 0.0405) in comparing the opinion of B.Com. students with the opinion of B.Sc. students, it became evident that there is a significant difference between these two degree groups at the 5% level of significance. In the B.Com. group 60.9% found the project useful, while 76.2% in the B.Sc. group found it useful (see Figure 4).

There was not a significant difference between students with or without prior expertise – both groups also found the project useful (p-value of 0.2819). 65.3% of the students with none, poor or average prior programming expertise found the project useful. It is pleasing to learn that 73.2% of those students with good and excellent prior programming expertise also felt that the project helped them to come to grips with the course content.

The students also do a practical module test for one hour, where they are expected to implement a program in Natural. The students were asked whether they found the module test easy or difficult. Only 27.8% percent of the whole group thought that the test was too difficult with no significant difference between the B.Com. and B.Sc. groups. However, while only 14.3% of those students with good or excellent prior expertise felt that the test was too difficult, 33.1% of those with a poor to average level of prior expertise experienced it as such. Because the course is aimed mainly at this last group, this indicates that the lecturers should re-evaluate the difficulty level of the module test.

7. CONCLUSION

The result of our study showed that the student profile differ from our assumptions regarding the ratio of English to Afrikaans students. It also showed that we should revise our policy of prescribing the textbook by Robertson. It also indicated that the difficulty level of the module test could be too high for our primary target group. Yet, the overall objective that was set out for the INF154 course was met. The ultimate test is whether the students gained enough knowledge to justify the course. The questionnaire asked the students to indicate whether they now know more than before they started the course. The results showed that there is not a significant difference between the B.Com. and B.Sc. degree groups (p-value of 0.9340) or between the poor and good expertise groups (p-value of 0.0840), and that the majority of students (69.8%) feel that they are now more knowledgeable than when they started out. Only 19.5% said that they were not, and 10.7% were indifferent. Ignoring those students who were indifferent (and faulty data), 78.1% felt that they are now more knowledgeable.