The Motivation of Students of Programming

Tony Jenkins
School of Computing
University of Leeds
Leeds, LS2 9JT, UK.
+44 113 233 5768

tony@comp.leeds.ac.uk

ABSTRACT

Students approach the study of computing in Higher Education in increasing numbers from an increasingly wide variety of backgrounds. In most degree level courses one of the first modules students will encounter is intended to teach them to program.

As the students become more diverse, so do their motivations for taking their degree. Anecdotal evidence from many institutions is that students are becoming more tactical, and will engage only in those activities that they see as contributing to an eventual highly paid job.

This paper describes an investigation into the motivations of students for taking a degree in computing, and for studying programming in particular. The results raise a number of issues for the teaching of programming.

1. INTRODUCTION

The teaching or programming (perhaps more accurately the *learning* of programming) is a problem. Instructors will be all too familiar with the struggles of new students as they attempt to come to terms with an area of expertise that lies at the very heart of the discipline. Most experienced instructors will have been faced with final year students approaching a project or dissertation determined to avoid any programming at no matter what cost.

Much has been written about the most appropriate language and paradigm to use to teach such students, and the trade-off between choosing a language for its pedagogical suitability or the extent of its use in industry. There is an increasing literature on innovative techniques to support introductory programming; suggestions have included the use of visual props [1], theatre, and even singing [7].

Even with these innovations, the underlying way in which programming is taught remains the same. A typical programming course begins with a consideration of the nature of the task, and then introduces programming concepts (in whatever language) in sequence. Students are expected to practice the use of the programming tools by undertaking exercises. This model is so widespread that it must be reasonable to assume that at some point, presumably in the past, it worked; it produced students who were competent programmers. It is evident that this does not happen any more. What has changed?

This paper describes an investigation into a possible source of change – the *motivation* of the students as they approach a first programming course. Three aspects of motivation are considered – the students' reasons for choosing their overall programme of study, their reasons for taking a programming module in particular, and their general attitude to their studies.

The results of this investigation show that students are studying programming for reasons (and with attitudes) that may well surprise their instructors. It follows that these instructors need to address the key issue of *motivation* in order to promote better, more effective, learning.

2. MOTIVATION

The motivation (or otherwise) of students is a key issue if they are to learn (a happy side effect for an instructor is that highly motivated students will probably be more rewarding to teach). To succeed in any academic task, the student must be motivated. They must want to succeed. This is especially true in a practical discipline such as programming – students must be motivated to spend time practicing, even when there is no explicit assessment credit available. John Biggs [3] neatly defines an instructor's motivational role as "getting the students to agree that appropriate task engagement is a good idea". In a programming module, an instructor has to persuade the students to agree to practice by engaging in writing programs. An understanding of the students' pre-existing motivation is vital if the instructor is to succeed in this.

Unfortunately, motivation is an abstract concept that is difficult to measure in any meaningful way [2]. It is possible to observe a person's behaviour and from that to infer their likely motivation, but it is never possible to be certain. Some general categories of motivation can be observed and identified, however, in an attempt to describe why a student might value learning.

Fallows and Ahmet [6] propose an informal list:

- the learner's desire to please the teacher;
- the learner's perceived need for the material being presented;
- the learner's degree of interest in the subject material;
- the philosophical values and beliefs of the learner;
- the learner's attitudes to the materials being delivered;
- the academic and career aspirations of the learner;
- the incentives and rewards that are expected to accrue from the learning.

ITiCSE 2001 6/01 Canterbury, UK

Clearly, some of these factors will be stronger than others (and some are unlikely in a Higher Education context – it is hard to imagine many students setting out explicitly to please their lecturer). The key, though, to inspiring students through truly *motivational* instruction is to maximise the positive effects of each of these factors.

Entwisle [5] describes three more generic types of motivation:

- *extrinsic* – the desire to complete the course in order to attain some expected reward;
- *intrinsic* – deriving from an interest in the subject;
- *achievement* – competitive, based on "doing well" and (sometimes) better than peers.

It is obvious that students motivated primarily in one of these three ways will have very different approaches to their studies. An extrinsically motivated student, for example, would probably do very little for which there was no summative assessment credit. A student with intrinsic motivation could be expected to read around the subject (even when there was no assessment), act more on their own initiative, and form their own views on the material they were taught. If achievement is the main motivator the student will adopt whatever strategy they believe will allow them to secure the best results in the form of the highest marks.

This second list omits "desire to please". While it is unlikely that a student will be motivated to please their lecturer, it seems quite possible that they would be motivated to please some other party – their family, perhaps. Biggs [3] calls this "social motivation". Social motivation might also include to some extent the fear of failure, a very strong motivator for some. Some students may be highly motivated because they do not want to fail and be a "disappointment" to some group whose opinions they value.

Students of programming will be motivated in a host of different ways (indeed, it is unlikely that any two in a class would be motivated in precisely the same way). Some may be interested in programming for its own sake, while others may simply see a programming module as something to be passed in order to obtain a qualification leading to a lucrative job. If an instructor understands, and addresses, the motivation of a class of students, it must be possible to provide a better learning experience.

An alternative view of motivation is the *expectancy-value* model (for example, [3]). Students must *expect* to succeed in their studies, and they must *value* the eventual outcome. These two factors are said to multiply rather than add – if either factor falls to zero there will be no motivation, the students will become dispirited and will not learn.

An instructor must understand why the students value the outcome of a programming module, and must ensure that they are able to expect success. The instructor must appreciate the factors that are affecting the motivation of the class, and must become a skilled motivator as well as a skilled teacher.

3. THE STUDY

For the present study four types of motivation are defined:

- *extrinsic* – the primary motivator is the career and associated rewards that will follow from the successful completion of the course;
- *intrinsic* – the primary motivator is a deep interest in computing (or specifically programming) for its own sake;
- *social* – the primary motivator is the desire to please some third party whose opinion is valued;
- *achievement* – the primary motivator is to "do well" for personal satisfaction.

A fifth category, corresponding to "null motivation" will also be used to accommodate cases falling outside these categories. Such "motivation" would be characterised by statements such as "I just want to pass".

A survey was carried out among first year programming students at the Universities of Leeds and Kent. The two universities recruit from much the same pool and have similar entry requirements. The students included in the survey had chosen a degree with at least half the time spent studying a computing discipline. Programming was to be a significant part of the first year for all the students, and the programming module was compulsory for all those surveyed. The survey was distributed in a lecture at the same point in the course at both institutions; the chosen lecture was the earliest possible in the course, before any significant academic content had been covered.

The students were asked two key questions:

- What one word best describes your reason for taking your degree programme?
- What one word best describes your reason for taking this programming module?

These questions were deliberately left open-ended so as not to confine students to selections from some list and hence influence their choice. The students were also asked to choose one of the following statements to describe their attitude to their progress in their degree:

1. I want to do well for my own satisfaction.
2. I want to do well to please my parents or family.
3. I want to do well to please my teacher.
4. I want to do well so that I will get a good job.
5. I just want to pass.

The first four of these statements correspond to the types of motivation categorised as achievement (statement 1), social (2 and 3), extrinsic (4) and null (5). It was expected that the final category, intrinsic, would emerge in the answers to the first two ("one word") questions.

365 valid responses were received, 226 from Leeds and 139 from Kent. The return rate (in terms of the number of students registered on the modules) was 70%.

4. ANALYSIS

There was clearly going to be a great variety of responses to the first two, free form, questions. To overcome this, the words given by the students were sorted and categorised to group synonyms and other combinations of words reflecting similar motivations. This then allowed a numerical analysis of the results using broader, more convenient, categories.

For the first question ("Why are you taking this degree programme?"), the categories that emerged were:

- *achievement* – the main motivator is success, for its own sake. Typical words were "satisfaction", "challenge", and "ambition".
- *aspiration* – motivations centering on some future goal. Typical words were "career", "job", "money", and (with commendable honesty) "avarice".
- *enjoyment* – the motivation is the enjoyment to be derived from the process of studying. "enjoyment", "fun", "stimulating" were typical.
- *learning* – words that indicate that the very act of learning was a motivator. Typically "curiosity", "enlightenment", "learning".
- *passage* – there was no clear motivation other than the acquisition of a degree being seen as the next step in education. "continuation", "progression", "parents".
- *programme* – the motivation lies in some aspect of the degree programme itself. Typical words were "consolidation", "relevant", and "technology".
- *university* – the main motivation is simply the opportunity to go to university. "beer", "independence", "Leeds" all featured.
- *don't know* – responses that indicated no clear motivation. Examples were "boredom", "confusion", and "stupidity".

A similar process for the question specific to the programming module produced slightly fewer categories:

- *career* – the motivation is to acquire a useful skill that will be relevant in some future lucrative career. Typical words were "career", "job", and "money".
- *content* – the motivator is provided by some or all of the module syllabus. Words used included "Java", "OOP", and "skills".
- *compulsory* – there is no particular motivation. The module is seen simply as a mandatory part of the degree course to be negotiated. "compulsory", "force", "required".
- *enjoyment* – the main motivation relates to the enjoyment of the experience of the module. "exciting", "fun", "enjoyable".
- *learning* – the process of learning something new is the main motivation. Typical words were "curiosity", "interesting", and "knowledge".
- *don't know* – words that seemed to indicate no clear motivation at all. Some answers were mildly surreal, including "elderberries". Other answers seemed to indicate that the student had no idea why they were studying programming.

The responses were classified under these headings to determine the dominant motivations (if any existed) of the students.

5. RESULTS

The results are tabulated below. The slight variations in the total numbers for each part of the survey are due to a small number of incorrectly completed or blank answers to some questions.

Table 1. Motivation for Degree

	Frequency	Percentage
achievement	13	3.50
aspiration	147	39.95
enjoyment	23	6.25
learning	133	36.14
passage	5	1.36
programme	30	8.15
university	6	1.63
don't know	11	2.99

Table 2. Motivation for Programming Module

	Frequency	Percentage
career	32	8.79
content	52	14.29
compulsory	175	48.08
employment	17	4.67
learning	73	20.05
don't know	15	4.12

Table 3. Attitude to Studies

	Frequency	Percentage
own satisfaction	181	49.59
please family	1	0.27
please teachers	0	0.00
get a good job	172	47.12
just pass	2	0.55
don't know	9	2.47

6. DISCUSSION

Two factors are clearly dominant in the students' motivation for choosing their degree courses (Table 1). The aspiration for some future gain, probably in the form of a financially lucrative career, is the most common factor, closely followed by the desire to learn. The closeness of these two values is perhaps somewhat surprising. Many instructors are now tempted to assume that the students are taking computing degrees largely (even solely) as a route into a career in the IT industry, but these results indicate that a significant proportion of students on these courses remain committed to learning for its own sake.

The low figures for "passage" and "university" are also of interest. Universities often try to sell themselves to potential students on the strengths of their surroundings or of the city in which they are located. This does not appear to be especially important to the students. Nor do many students see a university degree course simply as the "next step" after school.

These results indicate that a large number of students are *extrinsically* motivated. An almost equal number are motivated *intrinsically*, but this seems to be more to do with the process of learning itself rather than with an interest in the subject. It is, of course, hard to separate this satisfactorily from *achievement* motivation – it may be that the value derived from learning is closely linked with the resulting sense of "doing well".

The responses to the second question (Table 2) show that almost half the students saw the programming module foremost as simply a compulsory part of their degree. This is worrying. Most computing departments in universities would regard programming as a fundamental skill that underpins an entire degree. There is a danger here that instructors will find themselves addressing the 20% who are interested in learning to program, and will fail to motivate the rest of the class.

The low number of students pointing to career aims as a reason to learn programming is surprising, and somewhat at odds with previous work at Leeds [4] where students identified programming as a vital skill in demand by industry. This may suggest that students are now approaching programming from an ill-informed position; they do not now have a clear idea of the marketable skills they need to acquire. Certainly, the responses to the first question seem to indicate that this would be of interest to them, and thus an important motivator.

The results of the final question (Table 3) are striking. Two motivations are dominant as the basic motivations of roughly the same numbers of students. These represent *achievement* and *extrinsic* motivation. The slight dominance of achievement is surprising for instructors accustomed to "tactical" students.

There is little evidence from any question that any significant number of students is motivated by an intrinsic interest in the subject itself – a somewhat depressing observation. This is an area that must be addressed, perhaps as part of induction programmes in computing departments. Students will not learn about computing at all unless they are taught to (or otherwise come to) *value* the outcomes.

7. CONCLUSIONS

Programming is a difficult skill to acquire. It is best learned by practice and, if students are to learn effectively, some at least of this practice will have to be self-directed. An instructor's key role is to persuade the students to do this and thus to *motivate* them. They must be motivated so that they will engage appropriately.

These were students in the first semester of their first year (in the first week, even). Their attitudes will obviously change as they go through their course, and these changes will be the objects of further studies. Some students may discover a genuine interest in programming or computing and develop an intrinsic motivation. When they are better informed, some may come to value the learning of programming more, for extrinsic reasons. At the same time, of course, it is to be expected that the financial poverty of student life will make some more interested in future financial gain and less inclined to learn for its own sake.

An instructor cannot afford to assume that students in a programming class are motivated to learn to program. Students have a set of aims – an agenda – which means that they are in a programming class; it is not always the case that they want to learn to program, or even that they are interested in the skill.

There is perhaps an implication here for the choice of platform and language. Extrinsically motivated students aspiring to a lucrative career will demand to be taught those tools that are currently in vogue in the industry. Universities may have to accept that pedagogical issues in the choice of platform and language must be secondary to marketing concerns.

If the "tried and tested" approach to teaching programming was once effective (and is no longer so), the question of what has changed arises. One thing that appears to have changed, or at least differs from many instructors' views, is the *motivation* of the students. Instructors must now devise ways to address this.

8. ACKNOWLEDGMENTS

Thanks are due to Ian Utting for distributing and collecting the surveys at Kent, and to John Davy and Karim Djemame for tolerating interruptions in their lectures at Leeds.

Ursula Fuller commented on an early version of the survey. This work was carried out as part of a research project supervised by Ursula and Stefan Kahrs at the University of Kent at Canterbury.

9. REFERENCES

[1] Atrachan, O. Hooks and Props in Teaching Programming. Proceedings ITiCSE '98, Dublin, 1998.

[2] Ball, S (ed.). Motivation in Education. Academic Press, 1977.

[3] Biggs J. Teaching for Quality Learning at University. SRHE/OUP, 1999.

[4] Clark, M and Jenkins, T. The Expectations and Intentions of New Information Systems Undergraduates. Proceedings of UKAIS Annual Conference, York, 1999.

[5] Entwisle, N. Motivation and Approaches to Learning: Motivating and Conceptions of Teaching. In *Motivating Students*, Brown, S et al. (eds.), Kogan Page. 1998.

[6] Fallows, S and Ahmet, K. Inspiring Students: Case Studies in Motivating the Learner. Kogan Page, 1999.

[7] Siegl, E. Why Do Fools Fall Into Infinite Loops: Singing to Your Computer Science Class. Proceedings ITiCSE '99, Cracow, 1999.