
Choosing An Introductory Programming Language: Does SCRATCH Make Sense?

Rob Oliver

Christchurch Polytechnic Institute
of Technology.
Christchurch.
New Zealand
Rob.Oliver@cpit.ac.nz

Amitrajit Sarkar

Christchurch Polytechnic Institute
of Technology.
Christchurch.
New Zealand
Amitrajit.Sarkar@cpit.ac.nz

Abstract

At Christchurch Polytechnic Institute of Technology the Bachelor of Information and Communication Technology (BICT) degree has a compulsory Stage 1 Introductory Programming paper. The ability of the students is relatively diverse and some will never either intend to follow a career in programming or in fact be suitable for this type of employment. Teaching introductory programming concepts to a group such as this has always been a challenge. In the past the course has commenced with acquiring and developing logic skills and demonstrating these skills by developing solutions to everyday business computing problems using a flavour of Pseudocode developed by CPIT and the consequential Desk Checking of these solutions also using CPIT developed procedures. Students invariably get these activities wrong as a consequence of not being able to 'see the logic working' and not being able to experiment with various alternatives. The SCRATCH programming language has been tried this semester for the first time to try to overcome these barriers to learning and to provide a high level 'drag and drop' language set that students can experiment with easily.

Keywords

Programming, Language, Introductory, High-Level, First Year, SCRATCH, Logic, Pseudocode.

This poster paper appeared at the 2nd annual conference of Computing and Information Technology Research and Education New Zealand (CITRENZ2011) incorporating the 24th Annual Conference of the National Advisory Committee on Computing Qualifications, Dunedin, New Zealand, July 6-8. Samuel Mann and Michael Verhaart (Eds).

Introduction

During the first few weeks of the course students work both on their own and in small groups performing a number of activities involving problem solving skills and the application of those skills to developing solutions in SCRATCH. In some cases new programs need to be developed, in others existing supplied programs need to be modified. Most of the problems involve gaming situations. The underlying reasoning being that they learn the logic building blocks of software development in a 'fun environment' before progressing onto real world business applications in another language.

Methodology

Two groups of first year programming students were given their introduction into programming concepts by using a series of short lectures to explain programming concepts and using the SCRATCH programming language to deliver the practical aspect of these concepts. Designing and modifying games were used to demonstrate these concepts.

As a separate entity the same language was used to introduce business programming concepts and requirements.

Although the SCRATCH language is not entirely suitable for developing complex business problem solutions it does demonstrate the major logic building blocks that are required in all business computing solutions so it is a language that fits well into Computing and IT Education.

Results

To date, students find SCRATCH to be an interesting OO type language that allows the easy construction of

segments of logic by 'drag and drop', thus removing the necessity to enter laborious lines of text. The ability to run the constructed code without a compiler interface to check syntax means that the testing of the logic can be done as soon as the lines of code are assembled. At the end of the SCRATCH labs a Moodle multi choice test which covers topics such as "Several ifs," Nested Loops" and "Nested Calculation" was conducted. Each Lab stream had 30 students and 29 students passed from each stream. Stream A's average mark was 4.73/6 while Stream B scored 4.87/6. One student from each stream managed to fail the exam by scoring 2/6. Pass mark was 3/6.

Conclusion

Having only run this environment for the first time this year it is too early to draw any conclusions as to whether the student learning experience has been enhanced or the course success rate improved. This will have to wait until several iterations of the course have been run and statistical data gathered. However it has been noted that students seem to warm to the experience of being almost immediately productive.