Programming Goes to School

Nick Hines
BICT Student
CPIT
Christchurch

Chris McCarthy
CPIT
Christchurch
+64 3 940 7550
Chris.McCarthy@cpit.ac.nz

ABSTRACT

The "new" Digital Technologies curriculum in New Zealand Secondary Schools, now in its fourth year, offers Computer Science – Scratch, JavaScript and Python. Because of how the NZ Ministry of Education takes and manages student statistics, the numbers are not readily available but it is understand that the uptake in Computer Science (Programming) is not as high across the country as Digital Media (web development and multimedia) or Infrastructure (hardware, operating systems and networking). One reason given is calling it Computer Science instead of Programming or Coding. Certainly young women are not encouraged by this nomenclature. An American study shows that results can be improved to 50/50 female to male school students. This poster paper explores the lessons that can be learned from this US study to benefit the New Zealand environment.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education – *Computer science education, curriculum*

General Terms

Programming, Coding, Digital Technologies, High Schools.

Keywords

Programming, Coding, Digital Technologies in New Zealand High Schools.

This poster paper appeared at the 4th annual conference of Computing and Information Technology Research and Education New Zealand (CITRENZ2013) incorporating the *26th Annual Conference of the National Advisory Committee on Computing Qualifications*, Hamilton, New Zealand, October 6-9, 2013. Mike Lopez and Michael Verhaart, (Eds).

1. INTRODUCTION

Ever since Camp (1997) talked about the incredible shrinking pipeline of the decline of women graduates in IT, educational environments have tried to increase women in computer science education by making programming more exciting and more accessible (Repenning, 2012). Camp (1997) maintained that female students were losing interest in computing at elementary (intermediate) school age – if they ever gained the interest in the first place. Long before NZ high schools began thinking of whether students should learn IT – and when the only IT course was Keyboarding, which had, in those days, a high uptake from young female students, the University of Colorado (CU) began researching into the teaching of coding to local elementary school students. Using AgentSheets (see Image 1) CU studied the supporting of students with game design and simulation building through drag-and-drop interfaces.

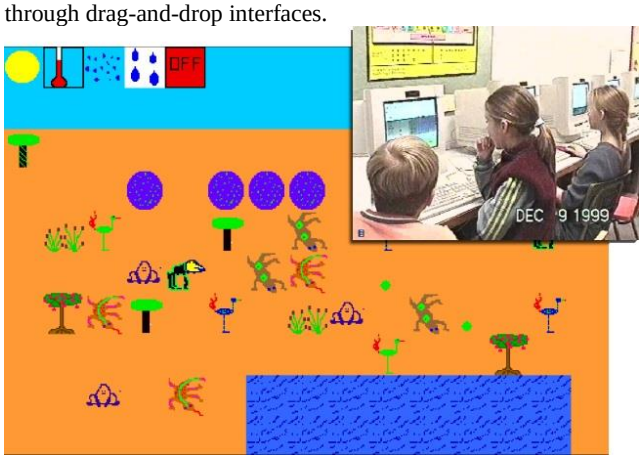

Image 1: Elementary school students learn about life science topics such as food webs and ecosystems by designing their own animals. Groups of students put their animals into shared worlds to study the fragility of their ecosystems.

CU had success with AgentSheets and later, AgentCubes (see Image 2) to explore the idea of 3D fluency. Because of their success almost 20 years ago, they became more concerned with why coding was not taught in middle (elementary) schools throughout the US despite such success with such tools.

From the CU research into activities, or lack of them, in other elementary schools throughout the US (Repenning, 2012) it was found that if programming was found at all, it was usually offered as an after-school activity. Middle school is deemed by both Camp (1997) and Repenning (2012) to be an important period in the life of young people, especially females, as they make many decisive but often unfortunate life and career decisions such as "science is not for me." Repenning (2012) asks how can we shift middle school computer science education from isolated after-school efforts from some to a full model where computer science is integrated into the school curriculum and taught in required classes at local and national levels?

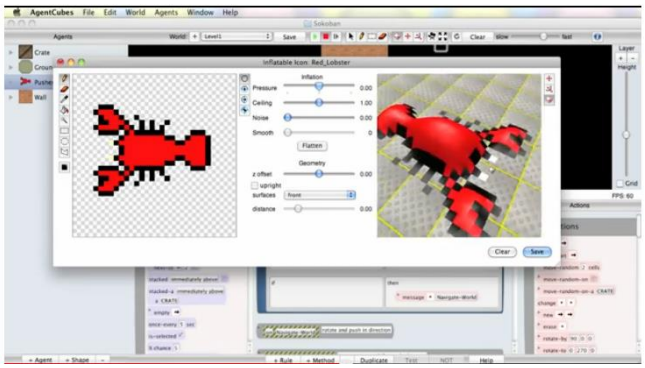

Image 2: Creating the creature (lobster) in 3D in AgentCubes

2. PUTTING IT INTO PRACTICE

CU's Scalable Game Design Initiative with over 8,000 participants throughout the middle schools of Colorado is showing great success. According to Repenning (2012) one of the Initiative's middle school teachers was using AgentSheets to introduce game design as a programming activity into his regular keyboarding and PowerPoint class. When the research team from CU visited this teacher's class they immediately noticed a completely different class composition. Instead of the single girl found in the computer club, 50% of the students were female. When the class was asked by the research team if they liked the game design component of the class, they all responded "Yes." When asked if they would also go to a computer club or after-school programming activity, most said "No." The universal excitement about game design in a required class suggested a strategy to increase the exposure of students' interest in computer science and broaden participation.

Since then the National Science Foundation (NSF) through its Innovative Technology Experiences for Students and Teachers (ITEST) programme enabled CU to expand this experience into a complete strategy for middle schools computer science education (National Science Foundation, n.d.). Studies were made on the effectiveness of this program in the way of teaching students computational thinking. The result being a set of advanced game design activities for middle school students, ranging from basic arcade games such as frogger (see Image 3) to far more advanced games such as the sims.

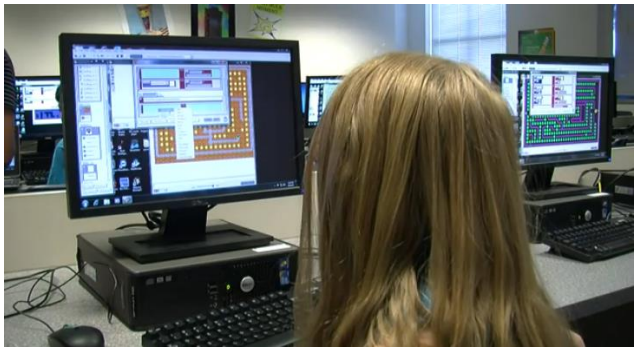

Image 3: Frogger, developed in AgentSheets by the young student herself, her about computer programming.

This finally led to the development of the Scalable Game Design Framework. The Framework has four main goals:

- Exposure - produce a highly adoptable and usable curriculum so a very large group of students were exposed.
- Motivation - motivating students to create a scalable set of game design activities ranging from low threshold to high ceiling activities.
- Education - build computational instruments that analyse student produced projects for Computational Thinking (CT) skills so that learning outcomes can be objectively measured (see Image 5).
- Pedagogy - systematically investigate the interaction of teaching approaches and motivational levels so that teachers could broaden participation.

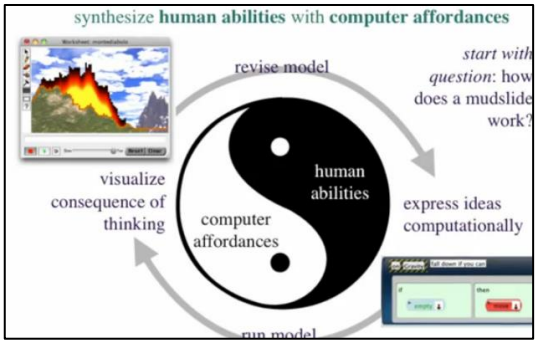

Image 5: the process of computational thinking (CT)

3. THE RESULTS

This certainly appears to be working throughout the state of Colorado and the results of dramatically increased young middle school female students being involved has come to the attention of the National Centre for Women and Information Technology. The Centre has supported and funded "AgentCubes in a Box" - a complete example curriculum that introduces Computational Thinking with the Scalable Game Design process! Students use the freely downloadable version of AgentCubes program to make a working 3D Maze Craze game.

4. CONCLUSIONS

This highly successful scheme has many lessons for the New Zealand education environment. Scratch is currently being used to some extent, but it does not have the game design and associated computational thinking as demonstrated by the AgentSheets and AgentCubes systems and the Scalable Game Design Framework. Certainly worth looking at as an alternative to Scratch in New Zealand schools – at any level.

5. REFERENCES

1 AgentSheets (n.d.) AgentCubes-in-a-Box: Introduce Computing Through Game Design retrieved on 15 September, 2013 from the AgentSheets, Inc website: http://www.agentsheets.com/

2 Camp, T (1997) The Incredible Shrinking Pipeline retrieved on 15 September, 2013 from the ACM website: http://dl.acm.org/citation.cfm?id=262813

3 National Science Foundation (n.d.) Innovative Technology Experiences for Students and Teachers (ITEST) retrieved on 15 September, 2013 from the NSF website: http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=5467

4 Repenning, A (May 2012) Programming goes back to school Communications of the ACM Page 38– 40, Vol. 55, No.