Guest Editorial Preface

Special Issue on Coding with the Raspberry Pi

Steve Goschnick, School of Design, Swinburne University of Technology, Melbourne, Australia

Christine Yunn-Yu Sun, eBookDynasty.net, Melbourne, Australia

In this issue we received numerous papers about coding and the Raspberry Pi, of which our reviewers have selected and helped improve three very good papers. They are representative of the current research around the Raspberry Pi in education. Each paper is focused on an aspect of teaching coding or ICT in three different geographic areas equally spaced around the world, namely: one in the UK, one in the US, and another in Australia. We also include two book reviews, both very relevant to coding on the Raspberry Pi.

FIRST PAPER

Developing Critical Understanding of Computing with the Raspberry Pi

This paper by Amanda Banks Gatenby, gives us a good insight into her research involving the Raspberry Pi over the last several years. The background educational scenario to her research is the change in UK school curriculum over the last 5 years regarding Computing; and the increasing usage of the low-cost platform that is the Raspberry Pi, by academics, teachers, students, hobbyists and makers.

The paper begins with a summary of the theoretical positions towards learning and criticality. Papert (1980) saw computing as an opportunity to make students aware that there were different ways or styles of thinking. As the start of a new literacy, no less. However, after seeing his Logo language usage appropriated in formal education within a standalone discipline, he was concerned his vision had instead materialised as a technocratic approach to learning. Specifically, he became concerned that an abstract, formal and logical style of thought, had come to dominate the educational approach.

Some educationalists saw the earlier single ICT approach in the curriculum as fostering a cohort of consumers rather than prosumers. The Raspberry Pi is a high-profile initiative by some entrepreneurial educationalists, to provide a generic tool that is fit for the purpose of reversing that situation.

The custodians of the UK curriculum chose to replace ICT with three discipline areas of Computing, namely: Digital Literacy, Information Technology, and Computer

Science. The author sees this replacement as a balkanisation of the curriculum, where it reinforces a dualism between academic abstract logical thinking, and an emphasise on vocational practices and making.

The author draws a parallel within the world of literature at an earlier time, but where critical thinking now has a broader meaning than simply the logical and abstract, that it once had. Where situated-ness and ways of knowing, can yield a multiplicity of answers to a problem over a single right solution.

The author sees the interpretation of the guiding curriculum by local educational practises, as an opportunity for pedagogical empowerment. The research reported on, set out to explore that opportunity by observing and measuring student perspectives, before and after set courses that used the Raspberry Pi - then a new technology to both student and teacher - in four different learning scenarios. Banks Gatenby was interested in a social theory of learning as the underlying principle, more so than the earlier theoretical positions.

Across the four scenarios she used the Q Methodology (Ramlos, 2016) to map the changes in perspectives, both individual and shared. The perspective in question is the student's perspective on the subject of *Computing*, and also of themselves as a student of this subject. Whereas the activities themselves were designed to build competence in Computing. The specific research question addressed is: 'How can the incorporation of Raspberry Pi technologies in learning design support development of learning citizenship and knowledgeability at school level?'

Q Methodology requires textural statements around the question being investigated. Thirty-four textural statements were arrived at via the relevant literature and a pilot study. Each study participant located themselves by sorting the 34 statements, from most through least with which they identify - both before and after the exercise in their particular scenario. With some factoring a normal distribution materialises.

The author analyses the results. Conclusions are drawn regarding what aspects of the learning space design (all involving the Raspberry Pi) influenced which changes in perspective, across the four scenarios. Does using the Raspberry Pi via practical exercises (constructing, done in teams), lean the teaching of computing in those scenarios, back towards Papert's goal of making students aware of "different styles of thinking"? Read it and see what you think.

SECOND PAPER

Local Area Network (LAN)-Based Digital Signage Solution Using Raspberry Pi

This paper by MengMeng Zhao outlines a case study in which a Raspberry Pi 3 miniature computer is employed as a Digital Signage System, that can be managed from either a laptop, a smartphone or a desktop computer. The local digital curriculum (VCAA, 2017) concentrates on Computational Thinking, with an emphasis on problem solving using information systems and technologies. The paper presents a case study that high school students (and their teachers) can follow, as a collaborative project that delivers a cost effective and flexible digital signage system for their school. One that can address a number of traditional school-wide signage problems. It presents an excellent use of ICT by configuring multiple interrelated hardware and open software, in a school community setting, that is a fit with many contemporary Digital Technology curriculums.

THIRD PAPER

High School Introductory Programming on Raspberry Pi Made From Scratch

This paper by Andy Luse and Bryan Hammer at Oklahoma State University, references a number of longitudinal studies that deal with the teaching of programming concepts to high school-aged students; in particular, with a view to increasing student self-efficacy. The authors set about describing an experiment run over a shortened period of time, with the purpose of increasing student self-efficacy. The overall game-plan is to increase the number of students that go on to a career in programming. The aim of the research was predicated on an earlier finding by others that: "…programming self-efficacy is found to increase performance, a key ingredient for developing interest…"

The experiment is made up of two exercises, the first is kinesthetic in nature (CS Unplugged, 2017), while the second involves the Raspberry Pi computer together with the Scratch programming environment, installed on it.

The sample size is not large, but with 22 subjects it is beyond just a qualitative study. The findings demonstrate a significant result backing up their claim that the approach raises the test subject's self-efficacy.

BOOK REVIEWS

There are two book reviews in this issue, both very relevant to coding on the Raspberry Pi, and particularly relevant to any researchers, educationalists, teachers and parents, that have a focus on creativity and coding in primary and secondary schools (K-12).

The two books are:

Lifelong Kindergarten: *Cultivating Creativity through Projects, Passion, Peers, and Play*
by Mitchel Resnick

and:

Objects First with Java: *A Practical Introduction Using BlueJ*
by David Barnes & Michael Kölling

Both books heavily involve programming environments available on the Raspberry Pi. Scratch has been on the Pi all along, however the later version 2 of Scratch was only ported to the Pi earlier this year. While BlueJ - a very innovative IDE (integrated development environment) for teaching Java and OOP (Object-Oriented Programming) - is also fully supported on the Raspberry Pi.

Resnick and his research group at the MIT Media Lab developed the Scratch language, which is the language that features most in the research studies in two of the papers in this Issue.

His new book *Lifelong Kindergarten* is not only about cultivating and supporting creativity in young minds, it draws wisdom from his 35 years experience at MIT, the last ten of which have involved Scratch and the eco-system around it, to push for a Creative Society. We have included a thorough book review and critique of this important new book.

You may have read or heard Mitchel Resnick (e.g. Resnick, 2013) and others emphasising that all kids should learn to code, indeed, learn to be fluent in coding. He doubles down on that good advice in this book, with:

Most people won't grow up to become professional programmers or computer scientists, but learning to code fluently is valuable for everyone. Becoming fluent, whether with writing or coding, helps you to develop your thinking, develop your voice, and develop your identity.

The second book reviewed is relevant to those involved in mid to senior high school, to that subset of students who will actually go on to be professional programmers and software engineers, or scientists and technologists, or video game creatives and entrepreneurs. Java and OOP represent an excellent second language choice, after the typical modern young student first learns Scratch or block-based coding somewhere in the K5 to K8 range. Java has been number 1 or number 2 language world-wide each year since the turn of the century, as measured in the metrics for such things (See: The Tiobe Index, 2017). You can program an Android with it, a smartwatch or a web server. You can use it as the intellectual fuel in your rocket or behind the scenes in your next animated movie or video game. The book and the BlueJ IDE that is intertwined with it, is all about learning OOP + Java. Both BlueJ and full-blown Java are available on the Raspberry Pi too, which is why we thought this second book was a good choice for a review in this Special Issue.

We hope you enjoy reading these three new excellent papers involving the Raspberry Pi. We thank the authors for submitting them to IJPOP, and we also thank the reviewers in helping to refine them. We think the two book reviews complement the papers in a significant way. It has been our privilege to be involved in putting this Special Issue together. We believe it will be of particular value to any academic researcher or educationalist, teacher or parent, involved or interested in developing creative learning and the creative thinkers of tomorrow. Enjoy.

Steve Goschnick
Editor-in-Chief
Christine Yunn-Yu Sun
Guest Editor
IJPOP

REFERENCES

Barnes, D., & Kölling, M. (2012). *Objects First with Java: A Practical Introduction Using BlueJ* (5th ed.). Pearson Education Inc.

CS Unplugged. (2017). Computer Science without a Computer. Retrieved Oct, 2017 from http://csunplugged.org

Papert, S. (1980). *Mindstorms: Children, Computers and Powerful Ideas*. New York: Basic Books.

Ramlo, S. (2016). Centroid and Theoretical Rotation: Justification for Their Use in Q Methodology Research. *Mid-Western Educational Researcher*, *28*(1), 73–92.

Resnick, M. (2013). *Let's Teach Kids to Code*. TED conference. Retrieved Oct. 2017 from https://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code

Resnick, M. (2017). *Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play*. Cambridge, MA: The MIT Press.

Tiobe. (2017). *The Tiobe Index*. Retrieved October 2017 from https://www.tiobe.com/tiobe-index/

Victorian Curriculum and Assessment Authority. (2017). *Victorian Curriculum: Digital Technologies*. Retrieved October 2017 from http://victoriancurriculum.vcaa.vic.edu.au/technologies/digital-technologies/introduction/rationale-and-aims