



## Sustained Excellence in Tertiary Teaching General Category

### Dr Richard Lobb

Lecturer in Computer Science  
Computer Science and Software  
Engineering  
University of Canterbury

**“My teaching philosophy revolves around transferring my passion for programming to my students. Historically, I have done so in person through lectures and tutorials, but today much of my teaching occurs through online systems which are designed to provide an experience that is both educational and enjoyable.”**

Richard’s journey as a computer programmer began in the late 1960s with the first mini-computer in the Southern Hemisphere and a soldering iron. The machine was funded by NASA, worth around a million dollars in today’s currency, and he was a Masters student in Auckland University’s Radio Research Centre. Richard, who was technically studying Radio Science, explored the capacity of this “amazing toy” and, after exploring hardware, quickly discovered that software was his passion. For his MSc thesis, he created a data recorder “that worked at lightning speed: it could punch 30 characters per second onto a paper tape!” He completed a PhD on analysing satellite signals, spending most of his time working with the computer.

The love for computers and computer programming has been pivotal throughout Richard’s professional life. Computers and computer programming were in their infancy when he left university and started programming at the coalface in London and Europe as the new industry blossomed in the 1970s. He never intended to be an academic, but did just that on his return to New Zealand in 1979. He was the first person to be appointed Lecturer in Computer Studies at Auckland University, albeit as part of the Maths Department. The best part of his job was sharing his love of computers and programming with his students so, when the demands of research and administration curtailed this, he retired from his position at Auckland in 2004 and moved to the South Island. Not quite ready to retire, he accepted a part-time, temporary contract to teach a new programming course at the University of Canterbury. Fifteen years later he is still there, doing what he loves – focusing on teaching and on sharing his enthusiasm for programming.

Richard has been instrumental in restructuring the teaching and assessment of the department’s classic first-year programming course COSC121 and for developing its new sibling COSC131, which had an enrolment of over 1,000 students in its first-ever semester in 2021. He coaches and mentors students, postgraduates and tutors across the department in a variety of ways. But the single most significant contribution he has made to the teaching of programming, not only at Canterbury or in New Zealand, but internationally, is the innovation CodeRunner, a programming teaching tool that is now used at over 2,000 institutions globally. CodeRunner is a free open-source question-type plug-in for the Moodle Learning Management System that runs program code submitted by students and gives feedback on their efforts. It provides a practice-based environment to develop programming skills and gives immediate feedback to students about successes and errors. It is adaptable to many different programming languages and is being used in disciplines beyond computing, including Mathematics, Engineering,

the Sciences, and anywhere programming is required, or where teachers are asking difficult questions that can be graded by a computer program. Since becoming an approved Moodle plugin in 2017, its usage base has grown from 94 registered Moodle sites to more than 2,000.

Richard believes effective teaching of computer programming is guided by the principal of “Learning by Doing”. He says computer programming is a skill akin to playing a musical instrument or riding a bike, where the only way to acquire the skill is by practice. Richard’s radical redesigning of foundational computer science courses has provided students with the opportunity to do programming instead of learning about it. Students respond better and learn faster if they are rewarded. CodeRunner provides immediate feedback on coding questions, which rewards students for their success at writing code rather than for their success in talking about it. During lockdown in 2020, UC’s course COSC121 Introduction to Computer Programming transitioned to remote learning seamlessly and successfully. Over 80% of students rated the video method as preferable to traditional lectures for programming-related courses.

Richard previously spent a lot of time lecturing huge groups of students in the largest lecture theatres on campus. He would use humour and stories of his early experiences in the programming industry in his lectures to enliven the presentation and was regularly nominated for the University of Canterbury Student Association Lecturer of the Year Awards and the students’ informal awards. But he was never convinced that this was the appropriate way to teach programming, which explains his enthusiasm for using videos instead. He is not nominated for as many student awards these days as he is not standing in front of large classes as frequently. But CodeRunner and the on-line model provides students with more options for learning and reaches far more students in many different disciplines and many different countries. Richard enjoys the many interactions with teachers around the world who are using CodeRunner.

In 2018, Richard received the Clinton Bedogni Prize for Open Systems for his outstanding contribution to Computer Science education (and other disciplines) through his development of CodeRunner. In 2011, he was awarded a distinguished UC Teaching Award and, in 2020, the University of Canterbury Teaching Medal in recognition of his teaching excellence.

Since 2006, Richard’s passion for programming has led to his involvement in programming contests. Student teams coached by Richard have often topped the New Zealand-wide contests and have then gone on to attend four South Pacific regional and three world finals. He ran weekly evening workshops for many years called Programming for Fun for students interested in competing. In 2011 he introduced a programming contest judging system called domjudge to Australasian contests and was a judge for the Australasian divisional and regional contests. He spent time as chairperson of the NZ Olympiad in Informatics, which runs training camps for high-school students competing in the International Olympiad in Informatics and has regularly taught a one-day segment of the training camp on the topic of Dynamic Programming. He has also taught a segment on Computational Geometry. Richard currently serves as Vice President of the South Pacific Program Contest Association.

“All teachers are limited by time and space as to how many students they can personally teach. My teaching programs break down many of those boundaries and allow me to have a hand in teaching the subject I love to students I have never met in places I have never been.”