Interview with Professor Rod Downey

November 13, 2007

Professor Rod Downey, `http://www.mcs.vuw.ac.nz/~downey/`, is a very well-known mathematician and theoretical computer scientist. He works in classical and applied computability theory and in complexity theory (mainly parameterized complexity), reverse mathematics and algorithmic information theory. Professor Downey is not only "world famous in New Zealand" (as a local saying goes in this country), but one of New Zealand's best known mathematicians. He has won numerous awards for his work in logic and his work in theoretical computer science. These include the inaugural MacLaurin Fellowship, the Hamilton Prize of the RSNZ, NZ Association of Scientists Research Medal, the NZMS Research Award, and the Vice Chancellor's Award for Research. He was elected a fellow of both the RSNZ and the NZMS. He is an editor of numerous journals, and chaired the prizes committee of the Association for Symbolic Logic. He gave an invited lecture at the *International Congress of Mathematicians* (2006), and has given invited addresses an numerous conferences including the *International Congress of Logic Methodology and Philosophy of Science*, and the *IEEE Conference on Computational Complexity.* He is also an accomplished surfer and Scottish Country dancer.

CC: Were any "mathematical genes" in your family before you?

RD: My family were really working class in Australia. I think I am the only person from either my mother's or father's side to ever go to University, and likely past 10th grade. My father who left very early was a lens grinder, and then a storeman and packer, but also a bookmaker on the side. I think that figuring out the odds so that you don't lose money needs at least a modicum of mathematical talent. But, of course, to realise such a talent you need the opportunities which my parents never had I guess. My mother also worked in the credit part of a large national company called Waltons, and she had to figure out repayments and the like. They did not really have very much respect for "University Types", and really could not understand why I did it until I got paid for it!

CC: How did you first notice your mathematical talent?

RD: I guess I was always good at mathematics, and my reports down through the years show this. Maybe I liked it as it needed less study than other subjects!

CC: Were you interested in Mathematical Olympiads?

RD: From the point of view of those doing it I think it's nice in that it encourages mathematics, and more generally science. My eldest son was involved in the final few of the Chemistry Olympiad, and had all that training. He did not make the team, but it was a great experience for him and encouraged him towards science. I believe that this is the really nice part of Math Olympiad, programming contests etc. It is such a good benefit to those who participate, at all levels not just those who go away as finally selected. The only negative I have is that it encourages a certain view of mathematics as combinatorial exercises, which colours the views of young people about the nature of mathematics. Do successes from the Math Olympiad go on to being good mathematicians? I don't know. But certainly one of my recent postdoctoral fellows, Antonio Montalbán, was very successful for Uruguay in the Math Olympiad, and is regarded as one of the finest recent graduates in logic in the world. He is now an Assistant Professor in the University of Chicago.

CC: Who was your first mathematical mentors?

RD: Hmm. What is a "mentor" varies with time. I have recollections about a collection of these. The first person who I remember really exciting me was a headmaster, called Harry Seldon, in a little school I went to called Rainworth with one class per year. This headmaster would take over from the other teachers at will, and try to inspire us with English poetry and mathematics. He was in his last years of teaching and had a very colourful past with many stories and "punishing" methods of class control, the cane being legal in those days. I still read and love poetry to this day, so maybe he had a big effect. I remember he challenged us by mocking "The Hollow Men" by T. S. Elliot, and later I went up to him and said that actually I liked it and thought it was a great poem. He berated me for not getting up in class and showing the courage of my convictions, a lesson which has stuck. He showed us the mysteries of Pascal's Triangle in year 6, and I was captivated by the beauty. He encouraged me to study my sister's high school book on Euclidean geometry and again I was fascinated at those wonderful proofs about inscribed triangles and the like. At high school we had quality mathematics teachers, but at some stage I decided that I wanted to do logic and philosophy, a high school subject in Queensland, usually done as an easy option instead of mathematics. Clearly as a logician many years later, it was a good option. We had a great teacher there, Miss Jenela Miller, who had real insight into logic, and encouraged me by loaning me university texts.

Later, as an undergraduate at university, I was inspired by the teaching of the then youngish Phil Diamond, Barry Jones (a wonderfully disorganised but insightful algebraist who clearly loved mathematics), and a logician Neil Williams, who challenged me with work by Tarski's student Szmielew on the decidability of the elementary theory of abelian groups. At Monash I was lucky to interact with the gifted Chris Ash, and a young student of Nerode called Rick Smith, who was very inspiring. But while he was kind of inspiring, one gift he made was asking me to prove a certain "theorem" which "ought to be routine".

For a long time I thought I must be so stupid as I could not get the proof to work, until I realised that the reason was that the theorem was not true! This lead to the discovery of a class of Turing degrees called the ANC degrees which I am still trying to understand to this day.

Finally I think there is that time when you meet someone who really opens your eyes to the really right way to think about mathematics, and how things “really work”. Often it is more than one person and to a lesser extend this has been true of my interactions with Richard Shore, Carl Jockusch and Mike Stob, but after talking with Ted Slaman I found my thinking had really changed and really opened up. The same was true later when I met Mike Fellows and his view of computer science.

Of course there are the other kind of mentors who you need I think. Those people who don’t directly work with you but whose encouragement and sometimes influence help so much. For me I would especially single out Anil Nerode, who has been unwavering with his support and advice, though sometimes I ignored him (you must come to America to succeed, etc.) Here in New Zealand Rob Goldblatt has been exceptionally supportive.

CC: Do you remember your first published paper?

RD: Yes. I had kind of two. The first was a research announcement, but the first real paper was published in the *Z. Math. Logik Grundlagen Math.* and I was so excited when the acceptance letter came on the recycled paper they used to use in East Germany. It wasn’t much of a paper, but I was finally a published author!

CC: As I said in the introduction you are one of New Zealand’s best known mathematicians. But you don’t work in “core mathematics”, algebra, analysis, geometry. . .

RD: That’s not completely true. It’s just that when I do things like algebra, I do them from a logical point of view. I have always been fascinated by computation and complexity. For example, one of my recent papers appeared in the *Journal of Algebra* and is devoted to understanding computable aspects of ideals in computable rings, and a recent paper with Montalbán looks at the complexity of the isomorphism problem for torsion free abelian groups. One consequence of this work is to show how badly the integral homology sequence is as an invariant for finitely presented groups. Logicians can attack questions like, when are no reasonable invariants possible for classification of objects? As a computability theorist, showing that such a problem is Σ^1_1 complete means that any set of invariants must be as complex as the classification problem itself, and hence no reasonable invariants are possible. I have been interested in effective mathematics for a long time, but have always had the view from logic. I guess that that is just taste. In computer science, the problems are so easy to state and so hard; it is so fascinating.

It is also true I think that one’s perception of “core” mathematics is kind of political. I think all mathematicians should know epsilon delta arguments, but by the same token, they should be forced to do logic.

CC: Very interesting. Do you present your papers in algebra to logic or algebra meetings?

RD: Almost always logic meetings, but I often use those papers as the bases for colloquia in math departments.

CC: Tell us about your work in reverse mathematics.

RD: Reverse mathematics in its modern form goes back to an ICM talk of Harvey Friedman and the efforts particularly of Harvey and Steve Simpson. The idea is that we try to understand the proof theoretical strength of various theorems of classical mathematics using calibrations in second order arithmetic. This idea really has its roots in the Greeks: is the parallel axiom necessary? The outgrowth is not only deeper understanding of these systems but also sometimes completely new proofs or, indeed, new mathematics. A great example of this is Montalbán's invention of totally new invariants (signed trees) to analyse the strength of various theorems concerning linear orderings. There are five basic systems coming from restricted comprehension axioms, saying that certain objects defined in a certain way can be "comprehended". The base system is called RCA_0 and is strong enough to prove basic facts, of, say, countable algebra including some strong results like "Every field has an algebraic closure", whereas to prove *uniqueness* one requires what is called ACA_0, arithmetical comprehension. Roughly speaking this is like comprehending the Turing jump. There are other systems, WKL_0 saying that every infinite binary tree has a path, and higher systems ATR_0 and Π^1_1-CA. There are also a bunch of currently poorly understood systems at the bottom, such as those related to Ramsey's Theorem for pairs. A nice example is the fact that the theorem that every countable commutative ring with identity has a prime ideal is equivalent to WKL_0 but if we replace "prime" by "maximal" then the result is equivalent to the *provably stronger* system ACA_0. The hidden fact is that there must be a different proof of the existence of a prime ideal to prove the prime ideal theorem. An excellent introduction is a classic paper of Friedman, Simpson and Smith in the *Annals of Applied and Pure Logic*: Countable algebra and set existence axioms. One must be careful with the results as they tend to be representation dependent.

It is natural for someone working in computability theory to consider reverse mathematics, since often there is an alignment of the effective content of a computable version of a theorem with the proof theoretical content viewed via reverse mathematics. Many techniques are familiar to a computability theorist. But it is a mistake to think that they are the same but clothed differently. A good example is the Robertson Seymour Theorem that finite graphs are well quasi-ordered by the minor ordering. This is true in computable mathematics, but not provable in *any* of the systems above even for graphs of bounded tree-width.

My own work has been mainly in algebra and combinatorics. With Stefan Lempp I investigated the well-know Dushnik-Miller result that every countable linear ordering has a nontrivial self-embedding using (for the first time) a priority argument to show that this is equivalent to ACA_0. I did some work on Hahn's Theorem and on free groups with Reed Solomon, and also with Lempp and Hirschfeldt on questions about extensions of partial ordering to linear orderings. (For example, every well partial ordering has a well ordered extension; this is a very strong theorem as it turns out.) More recently, with Lempp and Joe Mileti we look at the dumbest questions one might imagine. We showed that the "theorem" that every commutative ring with identity that is not a field has a nontrivial ideal is equivalent to WKL_0 but the "theorem" that it has a principal ideal is provable equivalent to ACA_0 and so provably harder! The same is true for vector spaces. Reverse mathematics is a really attractive subject, but I do think something could be refined here, such as the number of applications of the comprehension needed for the result. It is also related to proof mining I think, and I really like the work of Jeremy Avigad and his co-authors here.

CC: In your career what has given you the most satisfaction?

RD: Effecting change. I think I have really changed the thinking of a number of young people both in complexity theory and logic. It makes me really proud to see how many of them are succeeding all round the world. With the work of the NZIMA and the NZMRI I think we have been a truly positive force in New Zealand, with the brilliant Kaikoura, Nelson and New Plymouth conferences, as well as the joint Israel-NZMS and upcoming NZMS-AMS joint meetings. These were undreamed of when I came to New Zealand and I would like to think I was a positive force behind them. I also like to think I can ask good questions, and have effected change with parameterized complexity and with the current interest in algorithmic information theory. I have plans for future directions such as trying to comprehend the great similarities between online and computable thinking, but who knows? Likely most things will be done by the next crop of young people who seem to get smarter every year.

CC: What attraction can New Zealand offer to a famous Ausie mathematician?

RD: For me, New Zealand has been a great place to be a mathematician, and Victoria University has been exceptionally supportive. I came just before the WWW, and the internet. Early on, collaborative research was, sent a letter, wait a month for the reply (probably solving the problem in the waiting time), occasionally visit a famous place. This has changed with e-mail and web access. Also there was almost no money for research for mathematics in the 80's, but the Marsden fund has highlighted the world class work that some mathematicians have done in New Zealand and we are favourably viewed in New Zealand. Part of this is through the efforts of Vaughan Jones who, after winning the Fields Medal, worked on bringing world class people to interact with New Zealanders and summer schools etc., especially through the NZMRI (directors Marston Conder, Downey, Vaughan Jones, David Gauld, Gaven Martin), and later the NZIMA. Up to the present year I believe that we have had a nicer place to do mathematics than, say, Australia. But recently the demise of the NZIMA because of changes to government policy perhaps spells problems ahead, though mathematics is still supported indirectly by computational biology which is big here in New Zealand with the work of Mike Steel and others. The Marsden fund remains the main supporter and I have had a succession of very very talented postdoctoral fellows to work (and compete!) with down through the years. Working with outstanding young people is so much fun.

Also New Zealand has good surf and is really a nice albeit cool place to live. And I get to work in 10 minutes, and surf nice places. Life is too short to spend it in traffic.

CC: You are extremely prolific, publishing many multi-author papers. How do you organise this co-operation across continents?

RD: Part of the answer is above, via the internet. There remain co-authors I have not met. I have found mathematics and CS as a social endeavour. It is so much fun to work with others. That is the only downside of New Zealand, you really do have to travel to see others and this is nontrivial. I remember in the early days when I had little money, I asked the travel agent to find me the cheapest fare. This turned out to be Wellington-Sydney-Tokyo-Amsterdam-Tel Aviv, (to go to Haifa) and there were bad connections. 54 hours! But mainly if I start working with someone then I will continue this for quite a while after they leave and expect them to do the same. I have found also that short sharp visits you work up to are very effective.

CC: Tell us about your "parameterized complexity theory".

RD: Mike Fellows and I met at a conference in Palmerston North over some fine New Zealand wine. He had come to New Zealand to surf it being inspired by the old movie "Endless Summer". We discovered that we were both interested in complexity, something that I had been teaching myself at VUW. I had read his paper "Nonconstructive Advances in Polynomial Time Complexity" as was telling him about this stuff, saying it resembled his talk, before he told me that he was the author. He gave me the (Abrahamson, Ellis, Fellows, Mata) PGT paper (*On the complexity of fixed-parameter problems,* FOCS, 1989) the first attempt to study n^c vs. $n^{g(k)}$ asymptotic behaviour. The fact that there might be something interesting in this had been noted earlier in passing by Ken Regan, and also in 1982, Moshe Vardi had suggested that perhaps traditional complexity was wrong for databases, and a complexity based around the query size might be more appropriate. There were also some earlier work focussing on parametric issues without any completeness by Fellows and Langston in the mid 80's. This Fellows-Langston work was an important conceptual precursor. Of course, as can be seen in, say, Garey and Johnson, it had long been realised that "some forms of intractability seem better than others" but there was no real formalisation I think, except for approximation.

The PGT paper was the first to set up a complexity theory to try to explain the behaviour. The setup was nonuniform, only really applied to parameterizations of NP complete problems and the notion of reduction was extremely clumsy. Additionally, the hardness results needed something like "P-space complete by the slice." It was not fine grained enough also to address the final W-hierarchy. Still it is a seminal paper I think.

As Mike Fellows was well supported with loads of grant money, he asked me to visit him in Canada. We corresponded a lot, as this was the beginning of the internet and e-mail. Then at some stage, particularly on a surf trip around the North Island of New Zealand, we figured out the main definitions and proved some basic hardness results for the W-hierarchy, and some basic fixed parameter tractability (FPT) results. It was really exciting seeing it come out. Those first six papers we felt that we were really on to something. People like Anil Nerode were really supportive. I recently look at one of the first announcements of the material at Dagstuhl in April 1992. I remember giving the talk as the first of the conference maybe, and being totally jet-lagged arriving from New Zealand the day before, after maybe 40 hours travel. Someone at a recent Dagstuhl who was there said that at the time he thought that this was yet another uninteresting hierarchy in complexity theory, but he said to me how wrong he was. I found that comment very rewarding.

Practitioners of computer science will tell you that classical complexity often cannot be used to explain the computational behaviour of algorithms in real life. The idea is that we tried to design a complexity theory which perhaps better addresses this. The idea is to systematise the observation that sometimes when a parameter is fixed, perhaps a graph width metric in a hidden case, computational behaviour can be much better. The example always used is VERTEX COVER where for a fixed k, there are algorithms running in about $1.26^k k^2 + 2n$ for graphs of size n and these are implemented using simple reduction rules. They actually perform extremely well on, for instance, problems from computational biology, such as the work of Mike Langston on genomic sequences on irradiated mice. One the other hand, there seem to be problems such as DOMINATING SET where the only algorithm essentially "tries all possibilities". Thus it runs in time n^{k+1} or worse. The point here is that k-VERTEX COVER is linear time for each k whereas k-DOMINATING SET has the exponent increasing with k. Of course being able to prove that this is really the case would show that $P \neq NP$. Thus we use a hardness theory to show that things like k-DOMINATING SET cannot be done in time $O(n^c)$ with c independent of k unless there is a algorithm running in time $O(n^{c'})$ for deciding if a Turing machine with arbitrary fanout has a accepting path of length k, c' independent of k.

There have been many surveys of this area and now there is a whole conference devoted to it, as well as three books (Downey-Fellows, Flum-Grohe, Niedermeier) with another on the way (Fernau), and several Journal special issues (such as *The Computer Journal*), that I believe it has penetrated the consciousness to the extent that I won't develop this further here. (For details see my web site.)

What is really nice is that it has been developed by lots of strong young people such as Martin Grohe, Rolf Niedermeier, Vantakesh Raman, and their students, and the development is ever expanding. We know that you can use it to show no reasonable PTAS's and there are continuing interactions with things like ETH and non-approximations. We have even been successful in having real dialogue between people doing parameterized complexity and heuristics, such as the IWPEC and Dagstuhl series. It is really satisfying that our intuitions that this was something important turned out to be correct.

Actually early on we had a lot of negative comments about the work, reports saying that there was too much work in this area, and one classic saying "what this area really needs is for it to be developed by someone like (snip) (at a big name university like (snip))". It is hard to get new ideas to penetrate I think. The first book was important as it allowed people to access the material, rather then relying on "big conferences" so conservative. That is, rather than the social version of research which has us all working, like reef fish, on the current "hot" topic. I remember saying to Mike, if the stuff is good people will take it up. As anyone who knows him will say he always had absolute faith in the area.

This remains one of the nice things about Wellington, in that there is not much pressure on you to conform to the current fashions in research. I don't like the current CS culture of "what must I do to get into the next (STOC/FOCS/STACS/ICALP/CCC/etc)"; maybe this is a Math vs. CS view.

There is a huge amount to understand about practical computation. For example, we seem a long way from understanding why commercial SAT solvers are so effective when SAT is such a generically hard problem. As Moshe Vardi once said in a talk when he was young, SAT was thought of as being akin to bubonic plague (my paraphrasing), and now it is routinely used. The same seems true for commercial uses of automatic verification of properties. Perhaps there is some kind of parametric solution using the distributions. These seem such important problems to understand. Another example is the work of Abdulla and others using well-quasi-ordering theory to verify infinite state systems, but then implementing something that should take the life of the universe to work, at least in theory, but then on real data it works very nicely thank you. There is no theoretical explanation for this phenomenon. Surely if we understood why then we would be better able to design and understand such tools.

For the present time, I am willing to accept that $P \neq NP$ as a working axiom, but then this seems far from explaining all sorts of great questions arising from practical computation. Also, for instance, there have been astonishing results in complexity using the pseudo-random generators and it would be nice to develop this theory in the parametric setting.

CC: Can you explain in plain English one of your favourite theorems, its meaning, significance, and, maybe, relevance outside mathematics?

RD: Do you mean favourite theorem of mine or of others?

CC: Yes, favourite and not necessarily yours.

RD: From others, my favourites include the calculation of the number of derangements (the beautiful fact that e comes in), the fundamental theorem of calculus (!), Kruskal's Theorem that finite trees are well quasi-ordered by a minimal bad sequence argument, Cohen's invention of forcing to solve the Continuum Hypothesis, and Sacks' density theorem for the computably enumerable Turing degrees. I guess I am attracted to material by the depth of the ideas in the proofs. For my own, I would like to say one of the $W[1]$ hardness results, such as the early Downey-Fellows series, or the paper with Fellows, Geoff Whittle, and Alexander Vardy on coding theory, which says a lot about practical coding questions for linear codes, or the requirement-free solution to Post's problem with Hirschfeldt, Nies and Stephan, but to be honest the paper which sticks in my mind as most beautiful is one about nonembeddings in initial segments of the computably enumerable Turing degrees. To me this is such an elegant nonuniform argument. Likely it has little meaning outside the area, but the ideas of the proof are so nice to me.

CC: You are regarded as one of the world's best in logic circles. Please share with us your vision about the future of this important discipline.

RD: Logic is a funny game. There is classical logic, being set theory, computability theory, proof theory and model theory which historically reflect classical mathematics. In their classical forms to me that are important in that they give insights and alternative views of what mathematics is. For example, it is interesting that category theory gives one version of what a "natural" transformation is, yet in logic the idea of natural seems to go hand in hand with definable. Computability theory and complexity theory to me ask more deeply "what does it mean for there to be a solution". I was talking with one of Vaughan Jones' students who was studying "classical" mathematics, and found it hard to calculate various polynomials related to the Jones polynomial, and was totally unaware of what $\#P$ is. There seems now a trend that this classical form of logic is going back more to its roots and giving more insight back into classical mathematics. What is so sad to me is that because of hiring policies, there seems a loss of this expertise from many departments around the world. This is unfortunate as there seems no reason to limit your perspectives.

On the other hand, there has been a spectacular interaction between logic and computer science, with much of the world's best work in logic being in computer science. There are nice surveys in and around "On the Unusual Effectiveness of Logic in Computer Science" (Moshe Vardi). If you think about it this is really natural. For example, in parameterized complexity, and algorithmics more generally there has been so much emphasis on logical structure and metatheorems, rather than heroic case analysis. It is also a salutary lesson to us and maybe funding agencies. I remember when I was young a senior logician referred to fuzzy logic as "mathematical pornography", and now look how this has revolutionised the construction of, for example, washing machines. So much for intuition.

From my own point of view, there are some really brilliant young people in logic, both classical and applied, and in terms of results the future looks fine. Whether they will win things like Fields Medals or Turing Awards is not clear to me as those things are so political, and the committees full of people who think whatever they do is the hardest and deepest thing on Earth. I found it a real tragedy that Shelah did not win the Fields Medal, in spite of the enormous depth and breadth of his work. But then again, neither did Lovasz nor Szemeredi.

CC: In the last years you have intensively and extensively worked in algorithmic information theory. Tell us about your long anticipated (and already extensively cited) book *Algorithmic Randomness and Complexity* (with Denis Hirschfeldt).

RD: In many ways this book was your fault Cris. One of my Postdocs, Richard Coles was working with you in Auckland and had come down with a question you had asked. This was immediately after attending a series of talks by Lance Fortnow in our NZMRI conference in Kaikoura (which appears in a collection of talks on basic complexity in a volume *Aspects of Complexity*, (with D. Hirschfeldt, editors), de Gruyter Series in Logic and Its Applications, Volume 4, 2001, vi+172 pages.). Denis and I worked on this problem for some time, and then discovered this huge and rather neglected area with exciting interactions between computability, complexity, and randomness. We then made the possibly foolish decision to try to organise a mass of difficult to obtain material such as Solovay's wonderful notes, material from the Russian literature, and other unpublished material such as Stuart Kurtz's Thesis. Much of this material was really hard to find, and in some cases even harder to understand. The main point of some of these papers seemed to be to claim the theorem rather than to make the reader understand. We felt that we should present this in an accessible form, plus a few results we had recently proven, little imagining what was about to happen. Having got Solovay's and Stuart Kurtz's permission, we began the project, as usual thinking this would take two years ... Simultaneously there was an explosion of work with many new theorems and avenues for exploration found by a lots of gifted people, such as Andre Nies, Wolfgang Merkle, Jan Reimann, Antonin Kučera, Jack Lutz, Ted Slaman, to name a few, and then the book kind of took on a life of its own. In particular, I had some really gifted Postdocs and co-authors Denis Hirschfeldt, Joe Miller and Yu Liang who really discovered a lot. So here we are six years down the track, and 600 plus pages.

The main theme is what do levels of randomness and levels of computational power have to do with one another. Li-Vitanyi is a good basic introduction to the whole area of algorithmic information theory, as are other books such as yours and Ko's, but our book is really in depth concentration on randomness for real numbers and the relationship of this fact with computational power. For example, it has been shown recently things like Chaitin's Ω which are random but have high computational power are quite exceptional amongst randoms, and almost all randoms have almost no computational power. This requires careful explanation, so *buy the book!*.

The survey "Calibrating Randomness" in the *Bulletin of Symbolic Logic* gives a reasonable idea, as well as a couple of relevant papers of mine "Five Lectures on Algorithmic Randomness" and "The Sixth Lecture on Algorithmic Randomness." It remains a source of amazement to me that the material on lowness has turned out to be so deep. This is also true of the connections between initial segment complexity and higher levels of randomness such as 2-randomness. The most obvious omission is resource bounded material, but that must wait till later. Now the project is "almost" finished, as we have said no to any more additions. We plan the book to be available mid-2008, but as anyone knows the final checking takes forever. Especially with a meticulous co-author like Denis.

CC: Is mathematics relevant to computer science?

RD: Definitely and at many levels. There are the obvious algorithms behind many programmes, there is the obvious use of mathematics in design, the use in complexity theory, and many less obvious examples. Look at the work verifying safety and liveness properties using automata and wqo (well-quasi-ordering) theory, look at Kleinberg's work with Google, and small worlds. There are so many examples. On the other hand practitioners doing something like software engineering maybe don't see the obvious mathematics in the same way the people running helical CT scans don't see the mathematics and CS behind these things. This is always the trouble trying to support mathematics and/or theoretical CS; that somehow we must constantly be expected to prove to everyone why we are important, and underlay many of the main advances.

Of course, prejudices go the other way. CS has so many problems that are gifts to mathematics, but many traditional mathematics programmes completely ignore these areas.

CC: We have a strong group in theory in Auckland and we work hard to integrate ourselves: we teach theory courses, but also we "infuse" small theory chapters in applied CS classes. I will quote you a comment posted on a stage one forum by a CS student: "Most of us are doing or have already finished Maths 108/150, some are even doing level 2 Maths subjects. I understand Maths 108 is the minimum requirement for most Computer Science level 2 subjects. Just wondering what would be the best course of action for someone like us that hasn't taken a Math course yet. A lot of computer science students I know only care about the courses they're interested in and passionate about, and usually Maths isn't their passion. It just seems a shame to have to take up Maths when some of the best programmers I know get by fine without knowing it as extensively as 108/105. Please don't suggest a change of education provider. I am doing Maths myself, just curious and asking for a friend." What would you respond?

RD: Difficult question. I am sure that there are a lot of students in first year who think that they could design all of their degrees etc. Young people have very strong opinions. All I can say to them is that we believe that mathematics will enrich their understanding of CS and their view of the world. Students also benefit I think from doing philosophy and English and the like. This question comes mainly from areas such as CS where the university experience is seen more like an apprenticeship, rather than one which is a broadening experience. We also see the other side of the coin. We have students planning to be CS majors and discovering that university mathematics, especially the discrete, has little resemblance to high school maths, and is fascinating. These are often the very best students who finish up doing our LOCO (logic and computation, half mathematics and half CS) stream.

One of the other problems we have is that CS students often don't give mathematics their best shots as they are too busy writing code. When students come to me with complaints that their time is taken up with COMPXXX then I suggest to them that "so the reason you are no doing my course with a reasonable work load is that you are choosing to work on the one with an unreasonable one, is that correct?" This usually has the desired effect.

CC: You play tennis, badminton and table tennis. What else?

RD: So long as my knees hold up I will continue with them. I also swim, surf and teach Scottish country dancing and learn ballroom dancing. None of this seems to completely halt the onset of the permafat of middle age. Of course, it is a bit harder after the surgery (I had a "resection" of half of the gastrocnemius for the treatment of soft tissue sarcoma this year), but you should always struggle on. Actually, I write Scottish Country Dances and have a book with one more on the way. Devising dances is kind of like proving theorems, where you see flow through space.

CC: Are you children interested in mathematics or computer science?

RD: The oldest one came to University to be a chemist, but has been lured to the dark side of CS, and now plans a career in research CS. He finds it very exciting. He is very practically orientated and is horrified how little I know about (e.g.) Java. The younger boy is more attracted to design and art and plans a career in design or perhaps fine arts. He is definitely not attracted to mathematics. You don't really know CS until you come to University I think. Both are addicted to computer games of strategy.

CC: Thank you.