Computing Is Not a Spectator Sport: Rethinking How We Introduce Our Discipline to Students

Marian Petre

Centre for Research in Computing
The Open University
Milton Keynes, UK
m.petre@open.ac.uk

Abstract: This talk will describe *My Digital Life (TU100)*, a distance learning module that introduces computer science through immediate engagement with ubiquitous computing (ubicomp). This talk will describe some of the principles and concepts we have adopted for this modern computing introduction: the idea of the 'informed digital citizen'; engagement through narrative; playful pedagogy; making the power of ubicomp available to novices; setting technical skills in real contexts. It will also trace how the pedagogy is informed by experiences and research in Computer Science education.

1 Computing is ubiquitous

The world of computing has changed: it is increasingly ubiquitous. Computers are becoming part of numerous manufactured objects that populate our everyday lives. The digital revolution might still be young, but it has arguably brought about the biggest change in our lifestyles in the last two hundred years. Yet our students still often come to us with a vision of 'computing' limited to working with end-user applications such as Microsoft Office. This is myopic: with the potential to change our world every bit as much as genetic modification and nuclear power, ubicomp will influence all of us.

The Open University's response is *My Digital Life* (TU100), an introductory module designed to align students' understanding of the computing discipline with their experience of ubiquitous computing in the world. TU100 makes powerful concepts and capabilities available to learners as their first academic experience of the discipline, in order to engage their attention while introducing fundamentals of both computational thinking and technical skills. Students are given hands-on experience of designing, building and programming the small, ubiquitous computers that will become increasingly common over the next decade.

This talk will describe some of the principles and concepts we have adopted for this modern computing introduction that focuses on the concepts and consequences of ubiquitous computing: the idea of the 'informed digital citizen'; engagement through narrative; playful pedagogy; making the power of ubicomp available to novices; setting technical skills in real contexts. The talk will outline some of the challenges, experiences

and research that shaped TU100, and it will summarize early evidence of impact with students.

2 The informed digital citizen

At the heart of TU100 is the idea of the *'informed digital citizen'*: a person living in a technological society who is able to make informed choices about how they incorporate computers into their lives; a person who is aware of the benefits of computation, but also its risks. The module encompasses key issues in digital citizenship, such as ownership of data, privacy and security of personal information, and identity online and in virtual worlds. TU100 not only explores the potential of, and threats from, technology; but it also sets out to empower thousands of people with the skills and confidence necessary to use, create and shape digital technology.

3 Engagement through narrative

Any introduction to computing and information technology risks overwhelming newcomers with a colossal amount of disparate information. TU100 overcomes this by using a strong narrative thread which runs throughout the module: *our relationship with digital technologies*. TU100 engages learners in the issues through its intensely personal narrative, which begins in the familiar setting of the student's own use of computers and data, and expands gradually to encompass the student's immediate surroundings, friends, society and eventually considers how technologies are transforming our world. The course addresses timely debates including the threat of the digital divide and the use of computers for political and social activism, or for crime or war. TU100's material is deliberately wide ranging. It is not confined to computer science or IT; instead it demonstrates the power of digital technology to transform activities as diverse as composing music, conducting science and re-examining historical evidence. Yet TU100 is practical as well as conceptual; it engages students directly with the technology and opens the door to a computing degree.

4 Playful pedagogy

TU100 is designed in the tradition of efforts such as Alice [Al11, CAB00, CDP03] and Scratch [ML07, MRR10, Sc11], which prioritise hands-on demonstration of what computing can do, use scaffolding and libraries to give even beginners access to powerful programming elements, use programming to generate visible effects of interest to students, and hence captivate the imagination of learners in order to widen engagement in the discipline. The TU100 teaching strategy is that computing is *not* a spectator sport, and that learning to think computationally is inextricably about doing things. This strategy is embodied in 'playful pedagogy'.

Our experience in robotics for children [PHJ02, Ro11] and in studies of what children do on their computers after school [PB07] provided useful insights, not least into the impact on learning of motivational technologies such as robots and online social networks. Children in informal contexts learn by tinkering: examining and modifying existing artefacts to make new variants. They learn by trying things out. When they engage with a new environment, children go straight to the examples; they don't bother with tutorials, if they can have a conversation instead. Computing is a routine part of play for contemporary children; our challenge was to make play a routine part of learning computing.

TU100 is a hands-on, experiential module in which students construct and program a number of fun ubicomp projects. Students learn through a large number of guided activities incorporating step-by-step instructions, colourful illustrations, sample programs and audio-visual guides. They build and interact with tangible devices, they explore their usefulness and limitations, and they reflect on the risk of their deployment. Students share their experiences with one another and with friends and family, helping to build an informed digital society.

5 Making the power of ubicomp available to novices

In order to put the power of ubicomp into students' hands, we designed a toolkit aimed specifically at novice users: one that is robust, welcoming, amenable to mass production and cheap enough to give to every student. No tools were readily available for beginners to design and build ubicomp devices; toolkits for engineering and electronics students are on the market, but they are generally expensive, and require considerable skills in electronics and programming. Our solution is the *SenseBoard* (a small computer that is tethered to a PC using a USB cable) and the *Sense* programming language. Using the *SenseBoard's* 'plug and play' touch sensors, microphone, temperature sensor, motion sensor, light sensor, lights and motors, even a complete novice can build a huge range of projects – either those included in the teaching materials or those of their own creation.

'Doing things' in ubicomp requires some programming. Traditional computer programming languages require students to know a lot before they can accomplish anything interesting; *Sense* is designed to bypass many of the traditional barriers to learning programming (such as syntax and typing) and to empower students to try things, to tinker and play, and to achieve tangible results rapidly. Our previous research and teaching in robotics has shown that 'doing things' motivates students to learn fundamental concepts, algorithms, and disciplines.

Sense is a visual programming language that embeds textual instructions within graphical blocks, each representing an instruction. Programs are built by dragging and dropping onscreen blocks, thus avoiding many of the syntax barriers that can impede novice students, and allowing rapid development and experimentation. *Sense* is informed by empirical research in the Department of Computing into programming representations. *Sense* extends MIT's highly successful *Scratch* language, incorporating richer instructions, better debugging, internet connectivity allowing users to send and receive data, and full support for the *SenseBoard*.

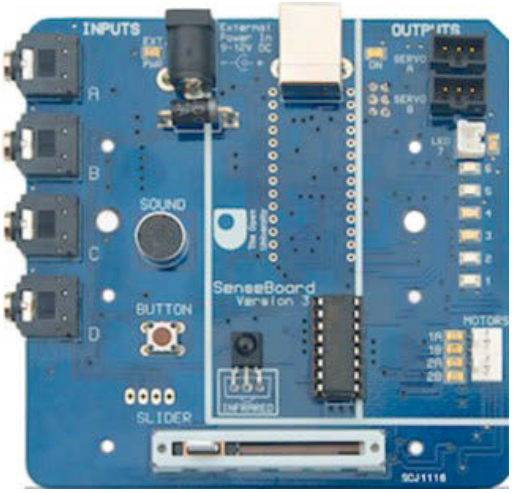

Figure 1: The SenseBoard

6 Technical skills in context

The development of technical skills and computational thinking are integrated into the discourse of 'informed digital citizenship'. The vast majority of projects demonstrate one or more applications of ubiquitous computing in an easily understood, controlled environment – yet one that clearly reflects the real world. Students are expected to reflect on the wider implications of their experiments. The TU100 learning materials include software tools that aid the construction of arguments and debates. We test their learning by asking students to contribute to discussions about topical issues concerning technology.

The issues are also considered in a global context. TU100 commissioned a large amount of high-definition video, giving students access to places and people that would not be available in any other university. TU100 students see advanced research being conducted by Cisco and Microsoft, see how a video game studio designs products, see the workings of a microchip foundry, and see interviews with the creator of the PC, digital literacy activists in Nepal and one of the activists behind Wikileaks. TU100 exposes distance learning students to world-class industrial environments.

7 Conclusion

TU100 has taken inspiration from childhood learning and commercial product design to produce compelling, yet academically rigorous study materials. The module appeals to students from all disciplines because the advent of digital technologies affects everyone, everywhere – even those who are unaware of the influence of computing in their lives. It does not presume any previous experience of the subject, but uses the big issues of ubiquity to motivate the acquisition of necessary skills: programming, computational

thinking, working with data, fundamental research skills, using online media to present arguments.

In its first 12 months, TU100 has attracted more than 5000 students, of which more than 70% were entirely new to the Open University. To the best of our knowledge, TU100 is the largest ubiquitous computing and IT module in the world, and one of the largest computing and IT modules in distance education.

Acknowledgements

This talk presents the work of the TU100 course team, led by Mike Richards and John Woodroffe.

References

[Al11] Alice, Carnegie-Mellon University. – URL *http://www.alice.org/* (08/2011)

[CAB00] Conway, M., Audia, S., Burnette, T., Cosgrove, D., Christiansen, K, Deline, R., Durbin, J., Gossweiler, R., Kogi, S., Long, C., Mallory, B., Miale, S., Monkaitis, K., Patten, J., Pierce, J., Schochet, J., Staak, D., Stearns, B., Stoakley, R., Sturgill, C., Viega, J., White, J., Williams, G., and Pausch, R.: Alice: Lessons learned from building a 3D system for novices. In: *Proceedings of the CHI 2000*. ACM, New York, 2000; pp. 486–493.

[CDP03] Cooper, S., Dann, W., Pausch, R.: Teaching objects-first in introductory computer science. In: *Proceedings of the SIGCSE '03*. ACM, New York, 2003; pp. 191–195.

[ML07] Malan, D. J., Leitner, H. H.: Scratch for budding computer scientists. In: *Proceedings of the SIGCSE '07*. ACM, New York, 2007; pp. 223–227.

[MRR10] Maloney, J., Resnick, M., Rusk, N., Silverman, B., Eastmond, E.: The Scratch programming language and environment. In: *ACM Transactions on Computing Education* 10 (4), 2010; pp. 1–15.

[PB07] Petre, M., Blackwell, A.: Children as unwitting end-user programmers. In: *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'07)*. IEEE, 2007; pp. 239–242.

[PHJ02] Price, B. A., Hirst, A., Johnson, J., Petre, M., Richards, M.: Using robotics for teaching computing, science, and engineering at a distance. In: *Proceedings of the 5th IASTED International Conference on Computers and Advanced Technology in Education (CATE '02)*. Acta, Calgary, 2002.

[Ro11] RoboFesta. – URL *http://www.robofesta-europe.org/britain/* (08/2011)

[Sc11] Scratch at MIT. – URL *http://scratch.mit.edu/* (08/2011)