Computing Education Practice 2019

Summary by J.H.Davenport

9 January 2019

Chapter 1

Session 1

1.1 Andrew McGettrick: Europe

Informatics Europe, ACM Europe and CEPIS. Background the ACM etc. report "Running on Empty" — what for Europe? Note also the US initiative CS4All. Initiative 30 January 2016, funding of $4G was mentioned. February 2018 "Informatics for All". Two-tier aproach: specialisation[1] and integration with other subjects. "Well-educated and skilled teachers are a vital requirement".

EU Digital Education Plan launched January 2018 separately.

Compares GCSE/A level (and scottish equivalents) for Computing and Mathematics. JMC produced "Digital Technologies and Mathematics Education". Looks at Pisa 2021 Mathematics Framework (especially paragraph 26).

Note also the recent CACM article about ML and education.

High premium on motivation of all sorts of students: top as well as bottom.

Q–PC Teachers tend not to be engaged.

A

Q There's no requirement for CS in school (Scotland), but growing interest at university.

A Real problems with Scotland.

Q There has been extremely popular "integration with business", but that's Excel.

A

---

[1]Casperson: "Learn to Compute" and "Compute to Learn".

1.2 Supervisor Recommendation Too for CS Projects: Terzić

Over 100 students (I know there's little sympathy for this). Rapid growth and I'm project coordinator. There's a list of 200 potential projects, but students aren't going to read this — how do we steer them?

Note that publication lists, module catalogues etc. exist, so some information is accessible. Use RAKE (Rapid Automatic Keywork Extraction) to produce keywords. Students can either input keywords, or a complete project proposal.

Matching algorithm uses synonyms from Wordnet to enrich matching process., based on semantic matching [Paoluccietal].

Backchecked against current year: got 7 right out of 12, but 10 out of 12 were suitable. Also checked against 32 staff-recommended proposals (not seen by tool), and in 50% of the cases it did suggest the actual author, and most cases was suitable.

The tool is not currently available — need to add a REST API, web interface etc. Then we can really explore.

Q How about assessing student suitability?

A Good question, especially for specialist MScs, where the dissertation must be on-topic. We currently leave this to the supervisors. Formal pre-requisites are done through staff project descriptions.

1.3 Role Choice of Female Students in SE Team Project: Laura Heels

Year 1 Core Programming

Year 2 Team Project — my focus. Teams allocated by module leader, but the teams decide roles. Teams with have ≥ 2 female (and international) or none. hence no isolation. Teams around 10 students. Typically 12 females in years of 100–150. Worth 20/120 credits (but they tend to over-commit, and see focus in terms of placement etc.).

Programmer as a role is often taken by females in year one of the tthree years studied, but not later. Peer Review occurs twice in the year. But programmers nearly always get the best marks in peer review. But all-male groups tend to call "secretary" as "organiser", and then it gets more marks.

Next steps:

1. Interview students for reasons for choice.

2. Look at relationship between role choice and career choice

3. Repeat in another university.

Q Why are there non-technical roles? IN Agile everyone should be responsible for everything.

A Doing a curriculum review, and the project may move to Agile. The original aim was to make people see the importance of non-technical roles.

Q Why year 1 difference?

A Possibly gender of module leader (female in year 1).

Q Is it really a free choice? Boys just decide that girls aren't technical?

A Hence importance of interviews.

Chapter 2

Session 1

2.1 Stewart Powell: Technocamps/Swansea

'Teaching Computing via a School Placement" — many students don't see the value of teaching. Note that there's a vicious circle, as poor tecahers give students a poor impression. Technocamps is a wales-wide schools outreach prgoramme, inspited by Faron Moller. Delivered workshops in 97% of state schools in Wales.

Also run an 18-day accredited teacher training course.

Y3S1 module (optional). Full day preparation in week 0: expectations, professionalism, what to expect. Assessment iscommentary, reflective blogs and teacher report (problems with moderation). Problems are finding schools (5/12 are welsh langauge, which is not the ratio of our students), finding free days in the students timetables.

Q How many?

A Currently 16.

Q Logistics, DBS etc.

A Yes, it's necessary, done prior to September start. Unlike PGCE, these will always have a teacher present

2.2 Henderson: Data Ethics

Aimed at MSc, but FY students also take it. A module description "talk about the crap in the news this week" wouldn't fly. I used "research leave" to get a law degree. See [Les09], which proposes code as one method: but Law, Norms and Markets are others.

9 bullet point syllabus. People have asked me for the syllabus.

Wanted to teach via seminars, which also got bounced. However, it has been successful. But need coffee and biscuits part-way through! [HPH$^+$18].

No books, lots of reading. But the students like the newness of the material. ut how do I evaluate? Are my students being more ethical?

2.3 Nurturing Collaboration

Software is built by teams, and emploers like soft skills. Talent supply an ongoing concern. Our current mantra is "doing it for real". Group projects have long been a feature of CS education [**?**]. Long a feature of Falmouth.

But just teamwork isn't enough. Look at "cognitive apprenticeship" as a model. Also "Legitimate Peripheral Participation" (scaffolding etc.). Teach agile from the beginning, peer review etc.

It didn't go very well, so initiated a Robot Olympics with Lego Mindstorm at start of first semester. Before/after was statistically significant. 43% of students have no programming experience, and very few women.

Q This is "just for fun" — assessment.

A Just ask them to reflect on it. But this was facilitated by 2nd year/faculty.

2.4 Papertian Mathetics with Concept Map Storied: Mandi Banks Gatenby

Module: "Digital Making and Learning: Theory" — mixture of F2F and DL students. This is not computation for its own sake. Papert asks us how we make "square". Papert asks whether this changes actually what we teach: new proof of Pythagoras. Use codefree programming, Greenfoot and Scratch.

The assignment is to produce a concept map (20%) but it's really about the evolution of the map: use a tool that allows looking at snapshots. Hence I can see how they're communicating with other course members. This forces a constant contact with me. Distance students feel more connected.

Q JHD — how many groups?

A It's individual. About 40 students. Depending on speed I do the history, it takes 15–30 minutes/student to mark.

Chapter 3

Session 3

3.1 Designing Security Assessments to Reduce Plagiarism: Roseanne English

I give my students an exercise, and they ask Google rather than look at my materials. Hence use less common exercises, less marks for "geting it working". 98 students, after three years of study BCS compulsory. Teams of 4, stages plus reflecive video presentation. Each stage output and process worth 4%, final presentation worth 18%. Each stage tasks has no easily adpatable code on-line.

1. Steganography. Told a file hidden, and size of file, but not told type, not told algorithm to find. Really struggled with "nothing online".

2. Decrypt cphered file

3. Explore with Java crypt library.

Video What happened and what you'd do differently. Strongly recommend an in-class session explaining how to do it. The students didn't realise that the first part, though worth 4% each, also fed into the video presentation. Took me a day to mark the video presentations.

3.2 Teaching Database Fudamentals: Lack of Progress Report: Boisvert

We've taught RDBMS for many years, and I have lots of past data (6 years and approx 70 students/year, mostly Busiess/ICT, with SQL their only programming language).

Out of 112 Nifty Assignments, only 10 use an important volume of real data (slowly increasing). Various experiments: adding YouTube, automatic SQL

grader, moving exam timing. [CampbellStanly1963: experimental and quasiexperimental designs]. I need to look at student engagement, not at improving my tools.

3.3 Data Protection: Glasgow

Hard to teach.

- Used US examples — "too remote".

- Used UK/EU examples — "too parochial".

- Used no examples — "too abstract".

Point is that the students have prior conceptions.

Divided students into 25 groups, each to research one country, and present results in one slide, 2 minutes. Before exercise: "EU good, Chna bad"; after "Germany good; Russia bad".

3.4

Greenfoot demo.

Interaction The character is not the right unit for editing programs.

Chapter 4

Session 4

4.1 JHD

See http://staff.bath.ac.uk/masjhd/Slides/CEP2019-Slides.pdf.

SF Veterinary Science "Day 1 contract".

JHD Indeed so.

4.2 Engaging with computer science when solving tangible problems: Charlton

"Everybody has to code" — why? "Fair exam — climb that tree" cartoon. [Chandrasekharan2009, Imaginations of movement]. Image "deep learning pit" of computer science.

Examples with Arduino's — positive feedback, but "very hard". Moved to Lilipad(?) and it was "why didn't we do this first".

4.3 JHD

See http://staff.bath.ac.uk/masjhd/Slides/XX10190-Slides.pdf.

4.4 Professionalism

Started in Freshers week, after for volunteers:expected 20, got 36. But only 7 submitted the (detailed) skills audit. Then 45' 1:1 to discuss skills audit. 4 students at focus groups. Wanted it to be student led, but also had someone from careers, and he really had to do most of the leading. But students got

the message about transferable skills: societies, local options, and one "putting back" in schools. They did "get the idea", but of course these were self-selected.

Next step is to work with second year students.

Chapter 5

Session 5

5.1 ?

Example — paying off a mortgage. What are admissible inputs/outputs? Write tests.

Emphasis is on steps of the process. This is harder to find on the web than specimen code.

First cohort finished last year. Three assignments: 15%, 35% and 50%. This sems to work for both computing and non-computing students. 09.2% Sat/Highly Sat — for course as a whole, not just this.

5.2 A Flexible Approach to Introductory Progamming: Gordon

Mixed background — hobbyists with bad mental models among others. 80+% male. Attendance poor: had students swiping friends' cards. Also Attendance $\neq$ engagement. See HEAcademy resource by Gordon.

Use SVN for seven years, but use it now to measure activity. Tutors can put skeletal code in student's copy. Test out of the SVN copy. We stream students in week 1, and give them harder problems.

5.3 Codio etc. Croft(+England)

Was 250 students when we started (rather more than I'd ever seen). I wanted formative tasks to be self-validating. Clear implementation issues. Hence Codio. Weve moved unit testing to week 2 of S2, and it's now an ILO.

Looked at varius, but Codio stood out. But Ethical Hacking, Games Progamming, it doesn't suit completely. And there people bit-mining etc. who

need policing. Saw a big drop in screw-up questions, and students mostly liked it.

Unexpected bonus was the utility of the data. Historical patterns of students doing work. Quasi-personalised e-mails based on the data has really improved attendance etc. Our biggest issue was regulatory over the university and GDPR.

Q Feedback?

A I give them the unit testing framework output, ME does more in terms of response for errors.

5.4 Business

Two meetings/year This is january, and a research meeting in September

Also Blog post, move the Twitfests to Wednesday.

Bibliography

[HPH+18] Michael Hessler, Daniel M Pöpping, Hanna Hollstein, Hendrik Ohlenburg, Philip H Arnemann, Christina Massoth, Laura M Seidel, Alexander Zarbock, and Manuel Wenk. Availability of cookies during an academic course session affects evaluation of teaching. *Medical education*, 52(10):1064–1072, 2018.

[Les09]   Lawrence Lessig. *Code: And other laws of cyberspace.* ReadHowYouWant. com, 2009.