

CEP 2020

Notes by J.H.Davenport

9 January 2020

Contents

1	BCSSession	2
1.1	Hayes & Nemetz: Bath	2
1.2	The PASS effect: Lesley May (UCLan)	2
1.3	3
1.4	Accreditation post-Shadbolt: Prickett	3
2	General	5
2.1	CodeRunner: Croft/England	5
2.2	Student Python Misconceptions, ? Glasgow	6
2.3	Errors with Git: Eraslan (Manchester)	6
2.4	Newcastle PBL Curriculum	6
2.5	DA Bootcamp: Beaumont (Aston)	7
2.6	Teaching Environments: DA Summer School: Glasgow	7
3	Afternoon	8
3.1	Engaging Science and Engineering Students (Newcastle)	8
3.2	Poster Presentations: Falmouth	8
3.3	Projects (Coventry)	9
3.4	Bradley	9
3.5	Brief Encounters	9
3.6	Increasing academic diversity: Lancaster	9
3.6.1	Diversity: Lynne Blair	10
3.7	UK SIGCSE AGM	10

Chapter 1

BCSSession

Been disseminating good practice via BCS website, but this session is a better way of doing this.

1.1 Hayes & Nemetz: Bath

2012: small department with one DoS: first in NSS. How to maintain this in the presence of growth? So appointed TFs, several DoS (one per year group), move placements from one person to a faculty team (80% do placements).

FN: Work with students representatives, try to grow community of international students. Female students now a group in BCSWomen and ACM women.

AH: placements. Post-placement is important: the returners are no longer students. Also post-placement conference feeds into students about to apply.

A

1.2 The PASS effect: Lesley May (UCLan)

PASS = Peer Assisted Study Sessions. Students are typically a bit introverted and this has really changed effect. We are a WP university (autistic spectrum) and historically no inter-year dialogue.

Students tell PASS leaders what they want to cover next week, which the leaders prepare. Weekly debrief of PASS leaders: “what’s difficult”. Or PASS is across the whole of first-year (common first year across all courses).

Safe place to ask questions, not necessarily about the course as such. The difference walking into the building is amazing: no more “small cohort” isolated groups. Before PASS, we couldn’t sustain student societies, but there are now Games development.

Q How did you discover?

A Course in Manchester

Q Does paying them help?

A Not a great difference in numbers, but but helps.

Q Participation?

A 70% of students attend at least 5 sessions. The sessions are timetabled.

1.3

OU Level 2 module. E-mail based game: aims for students to grasp the importance of prompt and effective communication within a function team.

Secenario Company with a glass building, and server room with a leak. 5 roles in the game.

Activity We have 900 students on module, 350 signed up for this game. Each student is assigned one role. There's an e-mail every day, but many of them are "office chat". Some e-mails have a real role, and various choices of responses.

Problem Students only see their role. Hence we now have a simulation of the whole activity.

The student feedback in the assignment based on this activity is good and shows that they have learned the importance of good business communication.

Q This looks like "adventure".

A Yes, there's a "write your own adventure".

Q Same assignment every year?

A Yes: as our students are very dispersed, there's little inter-cohort communication. But we have improved our management of the student expectations.

Q Does this affect their engagement with ITIL? ITIL is very dry.

A Seems to.

1.4 Accreditation post-Shadbolt: Prickett

Shadbolt is the excuse: we're trying to reach out to the community. There's a survey here. Recap of Shadbolt: recommends accreditation and industry involvement, also dissemination.

Note that we do look at %age of output graduation destinations, and to describe employability training.

Link to international standards, and a path to professional recognition. Trying to improve nationally the curriculum content, against pressures such as student opinion. LSEPI is still very relevant.

Q

A

Chapter 2

General

2.1 CodeRunner: Croft/England

We already had Codio (at the time wasn't secure for summative) in first year, immediate feedback etc. Additional checks such as memory leaks. But summative assessment was still exams (unrealistic) and coursework (academic offences and workload). Hence want a solution for summative. Told "can't use third party software", but CodeRunner, as a Moodle plugin, escapes this (why?). We can have multi-submission penalty attempts (have students with 60 attempts). Open tests and hidden tests (need this to prevent students hard-coding the answers).

In general, students are happy to be assessed this way.

- Students will game the questions (even more than you expect). Therefore don't have questions that return Booleans (will pass 50% of the tests).
- CodeRunner suggests all/nothing questions. This doesn't work for time-constrained summative assessment, else most students get 0.
- We're using C==14 and Python 3. Python 3 is much nicer, as C++ unit testing won't work against code that doesn't typecheck.
- Students need practice in this before the summative.

Q Partial credit?

A There's pre-check, which can be used for a range of things — up to test author.

Q Code quality?

A No.

Q Grades gone up?

A–ME Converse: replaces simple MCQs But course performance as a whole has improved.

2.2 Student Python Misconceptions, ? Glasgow

Python is syntactically simple (look at “Hello world” programs). But it is semantically simple.

'+' overloaded to float, but also string (OK-ish) and lists (no, it doesn't add the elements).

'+=’ Lots of web sites say “a+=c is short for a=a+c, but in Python documentation it's “augmented assignment”. The documentation is bizarre.

Test “Simple” list manipulation: less than 50% get it right. Replacing a=a+c by a+=c reduced confidence further.

JHD I teach second year Java and have the same problem. It's mutability, not the language.

Q Prior knowledge? Python common in England.

A Very mixed. Many from Eastern Europe have significant knowledge.

2.3 Errors with Git: Eraslan (Manchester)

Branching is a key aspect. A couple of related works. Difficulties of mixing Git with other command lines.

Problems when students create feature branches from others: “convention dependent”. We have a list of 27 items, 11 “convention dependent”.

A tool can check naming conventions etc., but “students do not make fetches” is not detectable.

Q Isn't this a “command line” problem? Can't we use tools instead.

A We are developing teaching materials around this.

2.4 Newcastle PBL Curriculum

Our degree didn't reflect changes in schools. Entry questionnaire in 2018 and 2019. This year 30% Python, 18% Java, 12% C. 73% has computing A, 73% had maths, and 11% had neither in 2019 (14% in 2018). What do you need extra help: 34% Essay writing, 32% time management, 20% Maths and 14% programming, which is not what we expected.

Old course had no real coherence in exam/practical weighting.

Process: student feedback (largely on assessment, as it happened), industry panel, all staff. Then created WGs for each topic, and overall steering committee. Partly try to unify tool use: had a “Chief Tool”. Emphasis on version control and continuous integration, so students leave with a portfolio.

New curriculum has 2/5 programming (1/5 Python, 1/5 Java) as 100% practical. Coursework informed by the theory taught in other modules. Yes 2 has

theory, then exams on this in November, then 100% team project. Then “Intro to specialist topics” (100% practical: paper writing etc.), and OS/Networks (50:50).

Stage 3 has all specialists in Semester 1, and research projects in semester 2.

Does it work? So far done One semester. A lot of technical issues. IT support updated GIT at the wrong time.

2.5 DA Bootcamp: Beaumont (Aston)

This has been running for some years, and we have graduates. CAP Gemini was our anchor employer. Now 350 apprenticeships with 13 employers. The CAP ones had done an L4 apprenticeship first. 4 intakes/year; 8–30 in each intake. Starts with an 8-week bootcamp, with 2FE delivering this. Aim to equi (mostly) school leavers with the core (technical and soft) skills needed. Thorough introduction rather than mastery of the language. Minimise time outside work. Teach OOP through Java (1 month), 1 week in business, 2 weeks DBM, 1 week final project.

Written exams (OOP and DB), oral exams “explain your design”, software portfolio. Progression is good (dropout is mostly non-academic). Some seasonality: September cohort is stronger (? motivation of later starts). Need to think about mentoring options.

2.6 Teaching Environments: DA Summer School: Glasgow

Also DA (Scotland), but lessons are more general. Range of prior knowledge, which interacts badly with lock-step delivery. Also want to raise the level of our computing courses. 8 weeks, optional daily two-hour drop in. Took first year students and gave them a programming test. Optional for those who passed (3 took it) mandatory for those who failed.

Adding test cases such as CodeRunner helped with the confidence, especially for those for whom a whole program was a big step.

Q Did all the students do all the material.

A All did all topics. It was individualised in terms of what sub-topics they did.

Q Dropouts?

A Some from degree as a whole. Also one who got married and adopted during the summer school.

Chapter 3

Afternoon

Plug for UKIE CER Glasgow 3–4. Abstract 5 June, full papers 12th.

3.1 Engaging Science and Engineering Students (Newcastle)

Students often have difficulty with the maths background here. Hence Physical Computing. Phase 1 — developing resources (using CS students). Phase 2 — select the best. Engineering etc. students watch tutorials at home before coming to workshops. Average mark was 2(i).

3.2 Poster Presentations: Falmouth

1/3 of time is group working in all three years, teams of 8–10, 2 of which are programmers (rest game designers etc.). There's a poster presentation: highlighting solution and their role in this. We chose posters as it forces interactive communication.

Guidance on posters seemed to be on font size etc., whereas we were marking for content. Also, because there's a lot of graphics in the group, the CS students feel overwhelmed. Marks poor, so did better Cue signalling, to tell them what we're really looking for. Ask them what their communication strategy was. Need way points. Students follow marks, so must emphasise the weighting. $5W + H$ [Miller2007].

42 students, median 19, 96% male. Effect size > 0.4 (Hattie's Hinge)

Q Variety of markers leads to a variety of marking scales.

A Pair assessing: one moderating and one asking.

3.3 Projects (Coventry)

WE have now separated teaching and module credits. The big problem with projects was student disengagement. So added continuous assessment (viva+individual marks, worth 33%). Semester 1 was 34% increase in assessment. There are still problems with consistency (or perceptions of this). Now groups present to each other. Also compact teaching to one lecture to all (saving moved to algorithms course). This has basically squashed the comparability issue. Need a solid mark scheme that makes sense to students and staff — showed current iteration.

3.4 Bradley

JHD missed this (IoC).

3.5 Brief Encounters

No science, all opinion.

Wanted: good communication skills, creativity, design literacy. We're not great at these. I give my students a brief, rather than a full specification. "Crits" are common in graphic design. No-one turns up: they're scared. Story boards are (just about) familiar with, but mood boards are new to them. This is important in my mobile application project: choosing colours and type faces. But this is final year: we need to do this much earlier. Also mark the process, not the product.

Q–JHD very difficult.

A I mark 20 students, but I agree the difficulty

3.6 Increasing academic diversity: Lancaster

IoC talk. We allow 40 credits of first year on a minor. Wanted to collect these. Genuinely CS, rather than IT. But needed to allow for scaling. Also difficulties of timetabling, hence blended learning.

Compulsory: Computation thinking and Programming (Python and JS) in term 1. Options in terms 2, project in term 3. Marketing video: "no previous experience, transferable skills, impact on society". Large recruitment (F2F rather than minors fair): 50% maths+stats, but a wide breakdown otherwise.

Studios: low/no monitors, to allow for interaction, cluster layout rather than rows. Whiteboards and breakout areas.

3.6.1 Diversity: Lynne Blair

Individual, department and discipline as three levels. Need outreach into local schools. Can we honestly say “no experience”, and if so how do we handle this?

SF CPHC is setting up an outreach network.

A

3.7 UK SIGCSE AGM

Elections: none. Finance: none. Note that Ireland has formally demerged, but still involved in CER.