This Teaching Cycle of the PGCTHE portfolio has been submitted to CADAIR with the permission of the author detailed above. It is to be used as a resource for future PGCTHE candidates and other staff as part of their professional development at Aberystwyth University. It remains the property of the author and Aberystwyth University. If you wish to cite this work then please contact the author. Contact details can be found at http://www.aber.ac.uk/en/directory/.
Chapter 2

Teaching development reports

In this chapter, I present my three cycles (or rather, three of my cycles, Section 6.1). The first one is about having interactive lectures (as stated on my poster, Section 1.3). The second cycle is about determining what students get from my lectures and, in particular, if they can relate the different parts. Finally, the third is about producing less intimidating slides.

I chose these three cycles because they cover very different aspects of my teaching. They do not apply to or concern all modules I teach but are representative of the kind of methods I test to improve my teaching. The numbering of the cycles does not reflect anything in particular, but their order in this document.

2.1 Cycle one: Interactive lectures

This cycle is about interactivity in lectures. As previously mentioned (Section 1.3), I always thought lectures in computer science were very static, involving almost no interaction with the students. The only interaction was when students were asking questions, which was not very often and, in some cases, even discouraged. At least, this was my experience as a student or as a person sitting in lectures out of interest.

However, as already stated in my essay (Section 1.1 page 6), teaching should involve students. An important part of that is involvement during the lectures, for example by encouraging questions (“Have you got any question so far?”). But in most cases, this is not enough as students often don’t ask questions (the I’m-going-to-look-stupid syndrome). I often tell the students that there is no stupid question, only stupid people (the ones you think the question is stupid), but that does not change anything.

1The two first modules I taught in the department contained the word “interactive” in their title.

2The first lecture of CS35310: INTERACTIVE MULTIMEDIA SYSTEMS consists in an introduction to the module, including a list of topics that will be covered. After that introduction, I ask the students whether there is anything else they want us to teach them (explicitly saying that an important part of the module is ‘interactivity’ and that they should use the opportunity). So far, no student suggested anything, but I do not despair. Actually, this year, a student in CS32110: INTERACTIVE COMPUTER GRAPHICS made a suggestion on his/her post-it (Section 6.1.1, page 67). Students also sometimes refer
To promote interactivity in my lectures, I regularly organise discussions about the topic currently being presented. I ask the students to discuss with their three or four neighbours about some subject. Students first discuss for about 10 minutes in these small groups, time during which I participate in the discussion of some groups, and then a discussion involving the entire class takes place.

The way I evaluate the success (or un-success) of these sessions is first by evaluating the level of participation, then the quality of the discussion and the ideas that are produced. The level of involvement is also important.

I have implemented that idea since my first lecture in Aberystwyth. I do that as often as possible, the exact amount depending on the module (very often — once every two or three lecture — in CS35310: INTERACTIVE MULTIMEDIA SYSTEMS, once every two weeks in CS32110: INTERACTIVE COMPUTER GRAPHICS, but never in CS21120: DATA STRUCTURES AND ALGORITHMS\(^3\). I am still improving the implementation of the idea and I use it in different contexts (theoretical discussions about some point of technical detail, discussions about ethical topics, design of programs, etc.).

Here is a list of problems I have had when implementing the idea and ways I solved them.

- Involvement is not general. Some students do not talk to their neighbours or some groups talk about the last football game (or whatever) but not about the topic. Because of that, I try to talk to most groups, or at least to the ones I know will not voluntarily participate. This has the inconvenient of, sometimes, prolonging the session for too long (as pointed out by Adrian Shaw, Section 4.4). However, I feel this is beneficial. If students stay alone, I talk to them and encourage them to join with some group. I usually never leave a group without guiding the students involved toward more discussion either on a side track or on more specific points. This is to make sure (or at least to try to make sure) the discussion in that group does not die as soon as I am away.

- After discussions in small groups took place, when a general discussion should take place, some students/groups don’t air their ideas. As a side effect of talking to all groups, I know who talked about what and where the interesting points are in the lecture room. I then guide the discussion using these ideas, prompting the students to air them. I tend to not force them, but sometimes do when there is a point in doing so (for example when it makes more sense when the student airs it).

The important ideas are usually addressed by the students because I guide small groups toward them when I talk to them.

- Collecting all the ideas is sometimes not easy, in particular when I want them to design a program. This for example, involves having the students to come up
to my lectures as interactive multimedia presentations!

\(^3\)This is mainly because I do not have a lot of time and because classes are large (120 students or so). Although I never tried, I also think that second year students would not participate as well as third year students. Obviously, this should change!
with a set of classes (this is object oriented programming jargon, sorry) to solve the problem at hand. Arriving at some general consensus or evaluating what the general consensus is is not easy. Next year, I will hand-out one post-it to each group and ask them to write down their design. I will then collect the post-its and classify them to get that global view.

2.2 Cycle two: Feedback about what students learn

An important bit of information I need from the students is what they get from my lectures. This is not just about factual information but more about the overall picture of the subject. In the modules I am more involved with (CS32110: INTERACTIVE COMPUTER GRAPHICS and more particularly CS35310: INTERACTIVE MULTIMEDIA SYSTEMS), there are many ideas students cannot pick if they do not integrate several lectures, relate different topics and, in general, think about what I tell them (see Section 3.4).

I cannot really ask them because their answer would be biased. The only moment I can get this information is when marking the scripts, but this is obviously too late.

I started having that need when, and possibly because of, discussions related to that were happening in the department. This was when Dr Mark Ratcliffe was working on MAP, the tool he (and others) developed to help with formative and summative assessment. However, that tool only allowed multiple choice questions and I wanted more general questions. I thus decided to try Blackboard, the application chosen by Information Services to “help” lecturers create a Web interface for the content of their modules. Blackboard also has an assessment part to it that allows setting questions like multiple choice, fill-in the blank, small essay, etc. Apart from the small essay, all the other question types had automatic marking and provision for feedback for students in case they failed questions. This was very appealing because evaluation of the students’ answers would not imply hours of work.

During the introduction lecture, I publicised the fact that I would provide a number of quizzes for the students to answer at the end of each part of the module. I also said that we would discuss their answers in the following lecture. And I did prepare questions and reminded them of that during lectures and by email.

From that experiment, I was hoping to get a “big picture” about what students understand, whether they can abstract general ideas and also push them to read my slides and the book to get more information as we proceeded, not one week before the exam. Getting this would mean that the idea was good. I was not expecting to get that at the end of the first part, but was hoping to see some improvement as we were making progress, as well as to see more involvement from the students during the lectures. So I did try to write my questions so that I would promote that, at first using mostly the type of question with automatic marking.

This has been a complete failure. Actually, maybe not, but I am not sure (see below). Unfortunately, I cannot include any data because, despite their assurances that this would not happen, Information Services are not able to give me all the statistics that Blackboard