Running Your CS Course

Ten Things to Think About

The applicability of this advice varies with course size and level. There is no right or wrong. Just think carefully about these aspects of course administration so that the ratio of learning to aggravation is maximized. For many more practical details, the CS undergraduate office maintains pages at http://eweb.cs.cornell.edu/coursewebsite.

1. Remind TAs about their Job Description
Tell TAs when to show up before the semester starts and tell them exactly how long they will be needed. (Up until the final course grading, the final exam, the final HW, the last day of class, etc.) In large courses it is good to have a TA attend lecture to pick up vibrations from the class and to be available for questions. If there is more than one TA then it is important to make sure that the load is evenly distributed subject to the 15-hour (PhD) and 8-hour (MEng) workweek upper bounds. Make sure they understand their role in the division of labor.

2. Assume Prerequisites
Be super-precise about prerequisites on the first day. Enrollees who disregard the prerequisites should not be allowed to torque what you do at the blackboard and they should not be allowed to strain the help resources. Be aware that some prerequisites are there just to peg technical maturity while others are there because of an explicit reliance on course material.

3. Post Assignments in a Timely Fashion
Even though many students ignore assignments until a few days before the due date, it is very important to hand out the work far enough in advance.

4. Guarantee the Efficient Return of Graded Material
Even in large lectures it is possible to return graded material after class if you have grader/TA help. It is chaotic for 5 minutes but very efficient. For a large course using administrative assistants to return graded material from their desks can kill productivity for days at a time. This can negatively impact other aspects of the department’s operation. Grading should be timely—assignment n should be returned before assignment n+1 is collected.

5. Pay Attention to Office Hours
Have regularly scheduled walk-in office hours a couple of times during the week. There is never a problem if these are occasionally missed as long as students have a mechanism for reaching you afterwards. Keep your administrative assistant informed about your whereabouts so that he/she can gracefully handle students who want to see you.

6. Carefully Keep Students Informed about their Letter Grade

---

1 Class lists, lecture hall technology, course handouts, exam schedules, photocopying, posting grades, room reservations, supplies, TA hours, textbooks, lab accounts, religious holidays, privacy issues, course websites.
Posting class lists with scores invites comparison and therefore promotes excessive competition and grade grubbing. Instead, give letter grade guidelines after exams/homeworks, e.g., $A = [90,100]$, $B = [75,85]$ etc. Deliberate gaps in the guidelines squelch attempts to grub for the single point that can turn a $B$ into an $A$. You are not obliged to post final course grades or cutoffs for $A$'s, $B$'s, and $C$'s. On the other hand, you can minimize borderline grade arguments by posting “approximate” cutoffs that are (say) 5 percentage points above the actual cutoffs.

7. **Oversee an Efficient and Fair Grading/Regrading Process**

Advise graders/TAs not to waste excessive time on incomprehensible solutions. Give these solutions a low grade thereby prompting the student to come in and offer an explanation. Put a definite time limit on when exams/assignments can be returned for a regrade. (Say, seven days.) This is especially important at the end of the semester. Students who do not respect the regrade window are sending a negative message about their interest in the course. Establish a way for students to review their final project/exam.

8. **Give a Final Exam/Project in Undergraduate Courses**

Final exams should be given in undergraduate courses unless there is some dominating final project that “sums up” the whole semester. The average student who takes a course with no final exam usually learns relatively little once the last prelim is taken. Remember that you are competing with instructors in other courses who use finals to capture student interest.

9. **Strive for a 100% return of Course Evaluations**

Have students fill out the evaluation on their own time after the last class. (A moot point if you use an on-line evaluation.) Force submission by using the threat of -5pts (say). This can be done in a way that does not compromise privacy.

10. **Appreciate Student Stress and Understand Academic Integrity**

First-time violators of the academic integrity code are often overstressed students who suspend their better judgement. Due dates that minimize collisions with other CS course deadlines, workloads that are relatively uniform across the semester, and the allowance of team submissions on programming projects all address the stress issue in a positive way. It is good to remind students again and again that it is far better to acknowledge an “illegal” collaboration at submission time than to have the staff discover an infraction at grading time. It is important to follow carefully the required procedures if you prosecute. Inform the undergraduate office before you contact the student(s) to prevent the accused from dropping the course before the hearing. As to penalties, there is a wide range of options. Written reprimands, zero scores, negative scores, half-grade reductions, and course failure are all defendable actions. Cheating on exams and submitting doctored regrade requests are more serious violations of the integrity code. Stay one step ahead by making sure your exams are well proctored; it doesn't hurt to be at the exam yourself. Discourage regrade cheating by loudly proclaiming that a random subset of the exams/assignments will be photocopied and that anyone who is caught improving a regrade submission will automatically receive a course grade of $F$.

---

2 More details at [http://www.cs.cornell.edu/ugrad/AcadInteg.html](http://www.cs.cornell.edu/ugrad/AcadInteg.html)