Negotiating the maze of academic integrity in computing education

Simon, *University of Newcastle*

Judy Sheard, *Monash University*
Amber Settle, *DePaul University, USA*
Andrew Petersen, *University of Toronto, Canada*
Chuck Riedesel, *University of Nebraska – Lincoln, USA*
Gerry Cross, *Mount Royal University, Canada*
Janet Sinclair, *University of Warwick, UK*
Michael Morgan, *Monash University, USA*

Why is academic integrity a maze for us?

- Academic integrity in programming is not the same as academic integrity in essays
- Guidelines for referencing in essays clearly don't apply to computer programs or various other assessments
- Very few people think it necessary to reference every line of code that might have come from somewhere else, even though that might be in conflict with the rules
- Programming is often a collaborative activity, yet assessment is an individual concept
- There are almost as many academic integrity practices as there are educators
- And they often bear little relationship to workplace practice

An ITiCSE 2016 working group

- Survey of academics and professional programmers
  - In what circumstances do they reference code or ideas that they might have found elsewhere?
  - In what circumstances do they reference assistance from others with code or ideas?
  - What explicit advice do academics give their students about academic integrity?
- And many more questions
- Then the analysis, brainstorming, etc

Some findings

- There is a profound lack of consistency among academics
- There are clear differences between academic practice and professional practice
- We wanted to propose clear guidelines that would improve
  - consistency
  - sense
  - usability
  - and more
- But we gradually realised that consistency is a false goal: different courses, at different levels, have different academic integrity requirements
- Our goal changed to one of helping academics consider and communicate these requirements

Research question 1

- What areas of current academic integrity practice engender confusion or conflicting views in academics and/or students?

- Reusing one's own code
- Taking externally sourced code and modifying it
- Getting help from others in various ways
- Attributing assistance and code reuse
- And more

Research question 2

- What are the differing perspectives and ranges of acceptable practices as perceived by computing educators and by computing professionals?

- All of the areas listed on the preceding slide

Research question 3

- What aspects of educational and/or professional practice in computing conflict with standard academic integrity policies?

- Design of code for reuse
- Reuse of the code of others
- Seeking the help of colleagues when stuck, and showing code to one another
- Whether and how to attribute externally sourced code

Research question 4

- Can we outline a process by which educators can design assessment tasks while taking into account the interaction of academic integrity, learning objectives, teaching practices, and professional practice in the computing industry?

- Not really . . .
- But we must justify specific academic integrity rules in terms of the learning objective of the task being undertaken

Research question 5

- Can we suggest a process that will facilitate the effective communication of academic integrity rules to students once these rules have been developed for a specific assessment task?

- A resounding yes

Proposals

1. For each assessment task, consider what constitutes acceptable academic integrity practice on various dimensions including attribution of code, code reuse, seeking aid, attribution of help, etc.
2. Link the defined acceptable practice to specific learning objectives and justify how these enhance the students' learning.
3. Explain to the students the academic integrity rules that apply to this situation, how they will be applied, and why they are justified in terms of learning objectives.

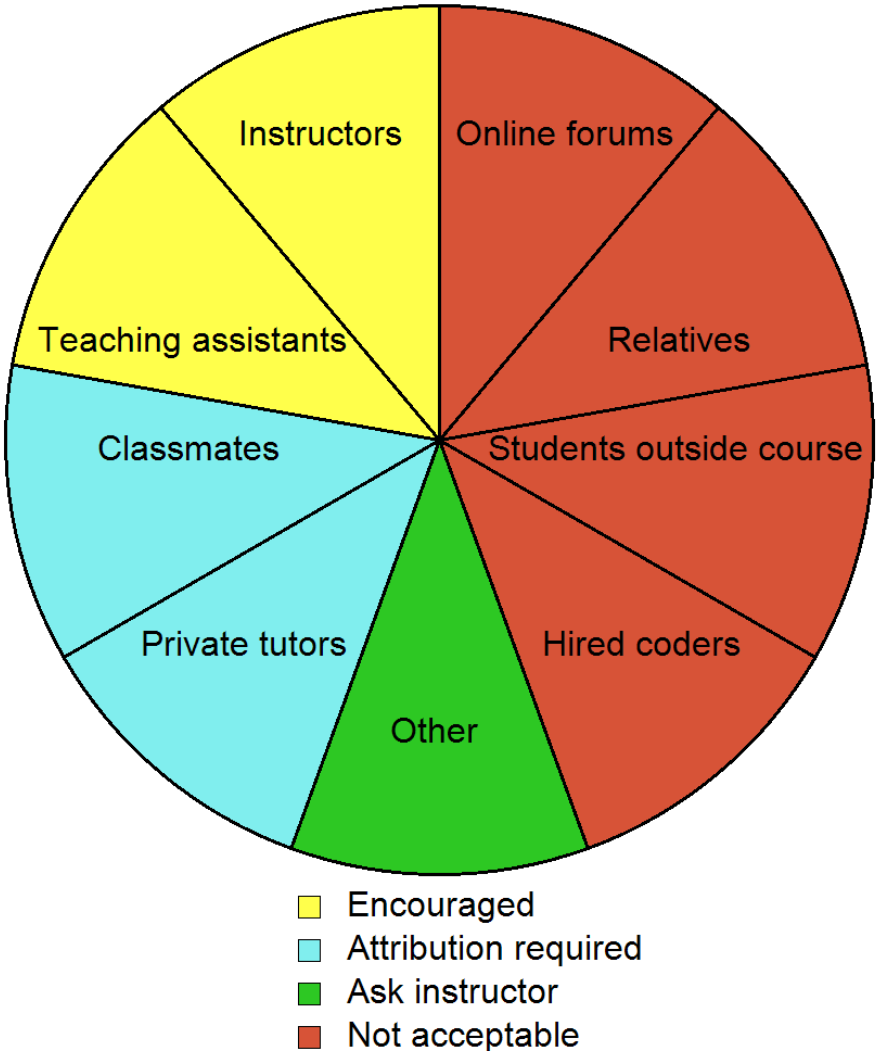
   (In the paper, we illustrated this using diagrams for relevant dimensions to clearly and simply inform students what actions and behaviours are expected, allowed, and disallowed in the situation.)

Sample guideline: intro programming

Assistance: Who can you get help from? Use this diagram to determine from whom you may seek help with your programs.

This assignment, which is to be completed individually, is your chance to gain an understanding of fundamental concepts of program structure and coding syntax on which later learning will be based. It is important that you master these concepts yourself.

Since you are mastering fundamental skills, you are permitted to work from course examples, but you must acknowledge assistance from other textbooks or classmates. In particular, you must not use online material or help from others, as this would prevent you from mastering these concepts.

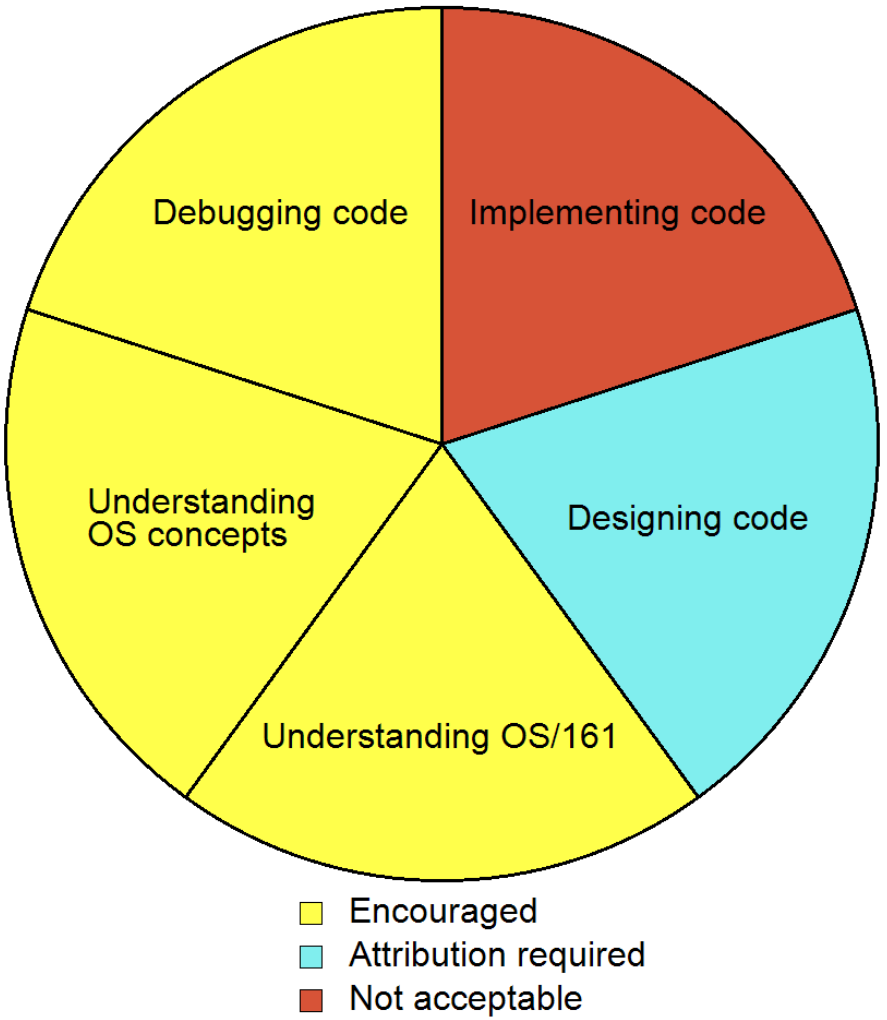


Sample guideline: advanced systems

Topic: What help are you getting?

These assignments, which are to be completed in teams of three, are your chance to demonstrate your understanding of OS concepts by implementing additional features in an existing pedagogical operating system (OS/161).

Since you are demonstrating your ability to implement OS concepts, you are permitted to collaborate broadly to understand the concepts being implemented, but you must be cautious when seeking help with designing your implementation or actually writing the code. In particular, you must attribute any assistance with the design of your code, and you may not, for this course, use code from other sources (or people) in your implementation.

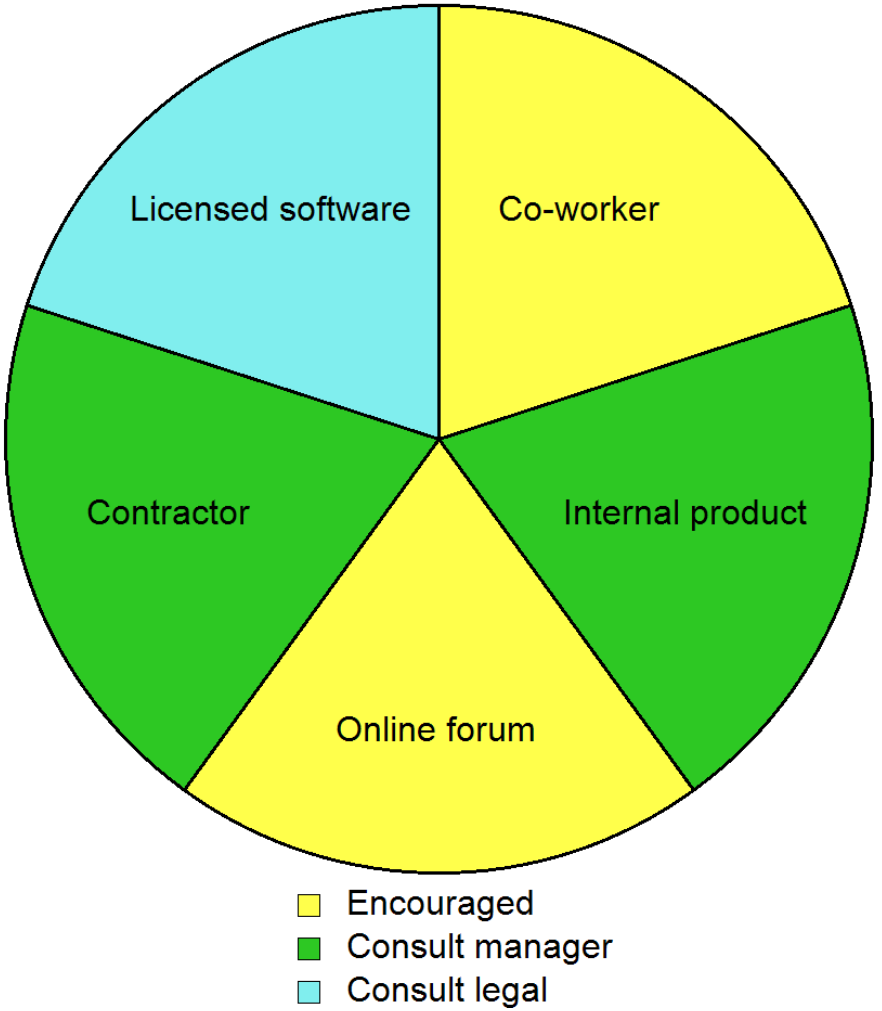


Sample guideline: capstone project

Source: How are you getting assistance?

This assignment, which will be completed in groups, is your chance to gain skills required to work in the computing industry. It is important that you work as a team and liaise with an external client.

Since you are practising workplace skills, you are required to work with project partners and may access external resources, but you must comply with standard industry codes of conduct and with legal requirements. In particular, you must not breach copyright or licencing regulations.



Conclusions

- Academic integrity in computing really is different from academic integrity in essays – which is what most university policies focus on
- Academic integrity requirements in computing vary considerably between different courses and different levels
- Students are often confused as to what is acceptable, which isn't surprising, because so are their instructors
- For every single assessment item we need to set academic integrity requirements in terms of the learning objectives
- And we need to communicate those requirements clearly and simply to students
- Of course some will still breach the requirements – but we can be more confident that this isn't through ignorance