A study of plagiarism

My students sometimes resort to plagiarism. As a university teacher, I have observed students cheat and plagiarize on a number of occasions. This study discusses the in depth reasons behind plagiarism, and suggests what teaching professionals can do about it.

This week a colleague discovered that a student handed in a programming solution with source code directly copied from the Internet.
My colleague showed signs of stress, and sadness, when she came to me for advice:

-*”Why did he do it?”*, she asked. *“And what is the process of reporting?”*.

-*”Just copy all code”*, I said. *”Zip it, together with the laboratory assignment instructions and send it to the disciplinary board email address. They will handle everything. You are not allowed to take direct action yourself”*, I told her.

-*”Now I have to spend time collecting evidence, instead of helping him”*, she said.

I sympathise with her, since I know she works overtime teaching this course and I know how it feels when one of my students cheat. It hurts and it feels like I have failed as a teacher.

The first time I encountered plagiarism I was a senior year student. That time I worked as a laboratory assistant in an senior level programming course. One of the international students handed in a source code solution with a lot of Swedish language comments. I confronted him and asked him if this was really his code, he quickly admitted that it was not. At that time, I thought it was stupid. But when I was taking a course in Digital Media Ethics I first learned that in some cultures, copying is not necessarily considered bad. Charles Ess, author of “Digital media ethics” explains that it can be considered an honor to be copied, and by copying, you are showing respect to the author(Ess, 2009). With this in mind, one must be even more clear with what rules we are using, when giving assignments to students from different cultures.

According to Tuba Gokmenoglu and Esra Eret, problems with clarity on what is considered plagiarism is a main reason behind it(Gokmenoglu, 2010). In their study they asked research assistants to answer statements with true or false. For example:

> “*if I read something and then write it as my own words; I will not plagiarize”* - (Gokmenoglu, 2010)

In 66.2% of the cases the research assistants answered that that statement is true. In total less than 80% of the research assistants gave the correct answer to all 11 statements (Gokmenoglu, 2010). No wonder the students plagiarize.

Last time I encountered plagiarism was in October 2012. I was giving a trivial programming course. Around 90% of the students pass the course every year. This year two students handed in code with striking similarities. We only discovered this by chance, since we happened to examine both students right after each other. When confronted, they admitted to have worked together, but said that they did not “copy-paste” it. I fear this case can be the tip of an iceberg, since plagiarism can be very hard to discover.

Detecting similarities in programming source code is not easy, especially if the examination is separated in time. This will change in the future since we are beginning to use new tools like Ticknik (ticknik.com, 2013). Ticknik can effectively detect plagiarism. I hope the tool will discourage many students from resorting to plagiarism, but I know it will not stop everyone. A better, more detailed examination procedure would ask the students to explain or change things in the source code. That way the focus can be shifted from solving assignments to learning. However, a more detailed examination takes more time.

Why will some students resort to plagiarism, even if there is a high chance of detection? One answer is: Learning programming is hard. This is due to the gap between what the students know and what they are learning is too wide. Learning programming is learning a new language, but also learning how to solve a domain specific problem. Thus the student needs to overcome multiple gaps to solve a single assignment. Imagine that you are trying to solve a problem, in a field you don't know, for someone you cannot speak to, and you can imagine how some of our students feel.

What can teachers do to overcome these problems? Daniel Coyle writes about deep practice as an effective learning strategy. Deep practice is achieved when we work on the edge of what we know. Coyle calls that the sweet spot, where we can make progress, even if we have to struggle(Coyle, 2009). Professor Sebastian Thrun sets a superb example with his MOOC course on Artificial Intelligence at udacity.com(Thrun, 2013). The learning gap between the assignments in Sebastian´s course, is very well arranged and focuses on either programming or the problem-domain. Also Thrun frequently provides inspirational lecturing.

Taking the time to creating an inspirational environment is something Ken Robinson speaks of.

> *“The real role of leadership in education […] is not, and should not be command and control. The real role of leadership is climate control, creating a climate of possibilities, and if you do that, people will rise to it , and achieve things that you completely did not anticipate and could not have expected”* - (Robinson, 2013 [17:58])

I have experienced just that. When students get inspired, they create a lot of positive energy that not only benefits the other students, but also us teachers.

Another colleague of mine, who spent time as a member of the disciplinary board of our university said:

> *“There are two main groups of students who cheat. The first group cheats systematically, and they are rarely caught. For the other group, cheating is a desperate action. They usually have tragic problems(money, relationships, gaming etc.) in their private life. “*

For the second group, is there really anything we can do? We must be sensitive and understanding, in order to find and help desperate students towards better solutions for their problems(rather than resorting to plagiarism).

Plagiarism affects both students and teachers negatively. Therefore we must do something about it. We do need better tools to detect plagiarism. But control is not everything! We also need to give our students better programming progressions and clear instructions. But most important of all: we need to create an environment that inspires the students to focus on learning! This takes more resources, but also creates positive side effects. If we could do that at my university, I'm convinced we would have much less plagiarism!

References

Coyle, D, 2009. The Talent Code. 1st ed. Bantam, pp.11-29.
Ess, C, 2009. Digital Media Ethics, Polity Press.
Eret, E, Gokmenoglu, T, 2010, Plagiarism in Higher Education, WCES 2010 http://www.academia.edu/235951/Plagiarism_in_Higher_Education_A_Case_Study_with_Prospective_Academicians> Accessed 2013-05-22
Thrun, S, 2013. Artificial Intelligence for Robotics, Website. URL: <https://www.udacity.com/course/cs373> Accessed 2013-05-10
Ticknik, Website, URL:<http://ticknik.com/> Accessed 2013-05-10
Turnitin, Website, URL: <http://turnitin.com/> Accessed 2013-05-22
Robinson, K, 2013. TED Talks. Ken Robinson: How to escape education's death valley, Movie URL:<https://www.youtube.com/watch?feature=player_embedded&v=wX78iKhInsc> [17:58-18:27]