Plagiarism and How to avoid it

Prepared by
Ruth Barrett
School of Computer Science
University of Hertfordshire

Costas Fostieris
And
Valy Pantiera
School of CS – IST Studies

Definitions

- Plagiarism is passing off someone else’s work as your own
- Collusion is working with someone else when you are meant to be working independently

Good Practice

- Refer to other peoples’ work
- Quote other people
- Reuse other peoples’ code

BUT BE SURE TO REFERENCE THIS PROPERLY
Bad Practice

◆ Pretending someone else’s work is your own
◆ Copying work without saying you have done this
◆ Using code written by someone else (including programs) without saying what you have done.

Citing other peoples’ work is valuable (1)

◆ Show the context in which your project is being done
◆ Demonstrate that you have researched the background
◆ Use ideas from others to develop your project
◆ You get credit for a good bibliography

Citing other peoples’ work is valuable (2)

◆ Back up your ideas by citing reliable published work

**BUT ALWAYS GIVE REFERENCES**

Failure to cite references indicates:
◆ Ignorance
◆ Attempt to deceive
Avoiding Plagiarism

Always cite source of:
- direct quotations
- paraphrases
- borrowed ideas & opinions
- borrowed diagrams and figures
- reused code

A common pitfall: errors in note taking

Distinguish between paraphrases and direct quotations
- Paraphrasing is putting the authors’ ideas in your own words. You must still cite the reference.
- Direct quotations go in italics within quotation marks ("direct quotation").
- Copy quotations exactly.

Example 1a (no plagiarism)

Direct quotation (reference correctly cited):
“The great efficiency breakthroughs in software are to be found in the fundamental architecture of the system, not in the surface design of the interface” (Tognazzini, 2003)
Example 1b (no plagiarism)

**Paraphrased text, no direct quotation (reference correctly cited):**

The fundamental architecture of a software system has a much greater influence on its efficiency than the surface design of the interface. (Tognazzini, 2003)

Example 1c - Bibliography entries

**Bibliography entry for examples 1a and 1b**

- Bruce Tognazzini, 2003, *First Principles of Interaction Design*

Example 2a (plagiarism)

**Direct quotation (no quotation marks, no italics, with citation):**

The fundamental architecture of a software system has a much greater influence on its efficiency than the surface design of the interface. (Tognazzini, 2003)

**This is plagiarism, since there are no quotation marks neither italics**
Example 2b (plagiarism)

- Direct quotation (no citation):
  “The fundamental architecture of a software system has a much greater influence on its efficiency than the surface design of the interface”.

- This is plagiarism, since the source is not identified.

Example 2c (plagiarism)

- Direct quotation (no quotation marks, no italics, no citation):
  The true measure of any compression scheme is the triple test of how well it compresses typical inputs, how quickly compression and decompression are achieved, and how much memory space is required for the method to operate.

- This is plagiarism, since no quotation marks, no italics and the source is not identified.

Example 2d (plagiarism)

- Mis-quoted text (reference correctly cited):
  “The real measure of any compression scheme is the triple test of how well it compresses input data, how quickly compression is achieved, and how much memory is needed for the operation.” (Witten, Moffat and Bell, 1999, page 406)

- This is Mixed Direct quotation and paraphrase. Only part of quotation is accurate.
Example 2e (plagiarism)

- Attempt to paraphrase that is actually mostly a quotation (reference correctly cited)

The real measure of any compression scheme is the triple test of how well it compresses input data, how quickly compression is achieved, and how much memory is needed for the operation. (Witten, Moffat and Bell, 1999, page 406)

- LESSON: EITHER QUOTE ACCURATELY or PARAPHRASE PROPERLY

Example 2f (plagiarism)

- Paraphrased text - no citation

To evaluate a compression scheme we need to know how well it compresses typical inputs, the speed at which it works, and the amount of memory space needed for the method to operate

- LESSON: ALWAYS CITE YOUR SOURCES

Example 2g (NO plagiarism)

- Paraphrased text - reference correctly cited

To evaluate a compression scheme we need to know how well it compresses typical inputs, the speed at which it works, and the amount of memory space needed for the method to operate

(Witten, Moffat and Bell, 1999, page 406)

- LESSON: By paraphrasing and properly referencing you demonstrate that you understand. You get credit for this
Example 2h (plagiarism)

Paraphrased text – reference correctly cited but in wrong place – at the end of the chapter

To evaluate a compression scheme we need to know how well it compresses typical inputs, the speed at which it works, and the amount of memory space needed for the method to operate

…followed by other material …… and far down the chapter …

……. (Witten, Moffat and Bell, 1999, page 406) ……

LESSON: Use references correctly

Reusing Code (1)

Plagiarism of designs and program code is taken even more seriously, because this material is supposed to come from you.

It may be considered as cheating, an offence with more severe penalties

Reusing Code (2)

Code re-use is often a sensible approach. Remember: it is one of the reasons for using Object Oriented methods!

Don’t re-invent the wheel: if a suitable program has been written, use it

BUT ALWAYS CITE YOUR SOURCE
Reusing Code (3)
Machine Generated Code

◆ Some development environments (e.g. J-Beans or Visual Basic) produce a lot of automatically generated code.
◆ Whatever the origins of the code you use, make sure you adopt a system for distinguishing that which is copied or machine generated from that which you have written yourself.

Reusing Code (4)
Machine Generated Code

◆ For example, you might put a comment at the start and end of each generated/copied passage.
◆ Do not remove comments that the development system generate in order to pretend that this is your code. This is cheating, an offence with very serious penalty.

Advice on using other's work

◆ Always cite correctly.
◆ Never pretend someone else's work is your own.
◆ Record all citation information, and construct your bibliography, as you go along.
Risks on Plagiarism

◆ Penalties vary:
  ■ From marking with Zero the offending work
  ■ To failing to the module with F3 (you will not allow to retake that module)
◆ Penalties are proposed by the FACO, and imposed by the Board of Examiners and vary between levels and cases.
◆ You can appeal to the Dean of the FEIS against the decision.