1 COLLUSION

Your lecturers will often set coursework for you that forms part of the assessment of a module. You will be expected to carry out the specified coursework in such a way that the markers can assess your abilities. If you work too closely with other students in a way that undermines our ability to perform the assessment then you have committed an assessment offence.

For example, if the coursework is specified as an individual piece of work and you work on this together with another student then this is the assessment offence of “collusion”. Similarly, if you are working on a piece of group work and you seek help from outside your group, then again you are colluding in undermining the assessment.

The University of Hertfordshire defines collusion as:

“evidence of the representation by an individual of work which he or she has undertaken jointly with another person as having been undertaken independently of that person; (UPR AS/C/6.1/ Appendix I)

However, there is a related offence to collusion. The University of Hertfordshire defines ‘other academic misconduct’ to include:

permitting or assisting another to present work that has been copied or paraphrased from a student’s own work without attribution or as if it were the work of the other.’

So if you help another student to do their coursework, then you are committing an assessment offence! This is fair; we need to assess the abilities of our students and if you impede that then you are regarded as having committed an assessment offence. Moreover, it is very unfair to other students if some students get unfair help.

1.1 HOW CLOSELY CAN I WORK WITH OTHER STUDENTS?

Lecturers will often advise students to help each other; it's a good way to learn, exchanging ideas, trying to explain something to someone else, and listening to the explanation of others. Students often say when accused of collusion: "but we have been encouraged to work together", and, "but I had to get help as I could not do the coursework otherwise"!

On the first point, when lecturers encourage students to help each other they mean with non-assessed work. The idea is that you do some work to help you learn, when it is OK to get as much help as necessary. Then when you have learned how to do something you prove it by doing the set assessed work on your own (or in a group for assessed group-work).

The answer to the second point above is obvious. Just because you cannot do the coursework doesn't justify you cheating, just as not being sure of being able to pass an exam justifies cheating in that!

So is getting help from other students with assessed coursework ever OK? In principle, the answer is no! It is also the safest policy to adopt. However, pragmatically, if you receive a small amount of help from another student it may be that the coursework markers will not notice or care. For example, if
you are trying to write a Java program and your friend says: "I'm going to use the HashMap class to store the customer records", and you think that a good idea and do the same then strictly speaking that is collusion. However, it is unlikely that it would get you into trouble if that was the only help you received. But if you received lots of help with the coding then you would be in trouble.

There is a very simple rule for determining the acceptable degree and form of co-operation between students: ask the lecturer who set the work!

1.2 WHAT SHOULD I DO IF ANOTHER STUDENT ASKS ME FOR HELP

The short answer is to say no! If you don't, you risk being accused of an assessment offence and losing marks. And remember; the penalties for assessment offences get much more severe after the first offence.

1.3 ARE THERE ANY OTHER PRECAUTIONS I SHOULD TAKE?

Yes; recall that the University of Hertfordshire defines ‘other academic misconduct’ to include: permitting or assisting another to present work that has been copied or paraphrased from a student’s own work without attribution or as if it were the work of the other.’

So, say when you are doing your coursework you leave copies of your work around in the form of paper, computer discs, memory sticks etc., and another student gets one of these and bases her or his work on it. Is it fair that you should be penalised? Well consider what would be the effect if lots of students did this? It would mean that we couldn't assess the abilities of our students by coursework. So it is fair for the University to penalise such students who do not exercise a reasonable degree of care.

The safest course of action is to be as careful as possible that your coursework doesn't fall into the hands of one of your fellow students.

2 PLAGIARISM

It is often necessary when writing reports or other documents to make use of the work of other people. This may be because you wish to lay the foundations for your work before describing it in detail, or it may be because you wish to demonstrate that you have taken account of the ideas and opinions of other workers. However, if you do make reference to the work of others you have a responsibility to make clear to your readers what is and what is not your work. If you do not do this then you are guilty of the assessment offence called plagiarism. The University of Hertfordshire defines plagiarism as:

“the representation by an individual, whether intentionally or otherwise, of another person’s work as their own or use of another person’s work without acknowledgement including:

a the importing of phrases from another person’s work without using quotation marks and identifying the source;
b making a copy of all or part of another person’s work and presenting it as the student’s own work by failure to disclose the source;
c without acknowledgement of the source, making extensive use of another person’s work, either by summarising or paraphrasing the work merely by changing a few words or by altering the order in which the material is presented;
d the use of the ideas of another person without acknowledgement of the source, or the presentation of work which substantially comprises the ideas of another person and which represents these as being the ideas of the candidate;”

(UPR AS/C/6.1/ Appendix I)
An important thing to note about this definition is that it says “whether intentionally or otherwise”. So even if you copy the work of another person unintentionally or you apparently claim credit for it because of your incompetence in not following the rules described below you have still committed an assessment offence. You may be surprised that unintentional plagiarism is an assessment offence but it is fair. If you make a mistake that could result in you being credited for work that is not your own then you have misled the person who has to mark your work. Consequently, there is the risk that you will be awarded an unfairly high mark!

The University, as do all universities, regards plagiarism as a very serious academic offence. There are two reasons for this. Firstly, in the academic world there are clear conventions and rules about these matters and the University as an academic institution is obliged to teach these rules to its students as part of their induction into the academic world. Secondly, the University needs to assess its students. This places an obligation on students to make clear to lecturers marking their assignments what is, and what is not, their work.

2.1 USE OF DIRECT QUOTES

A common way that students commit plagiarism is by using phrases taken from a published source without proper acknowledgment. Proper acknowledgement has two requirements: an indication of the extent of the quotation and an indication of the source of the quotation. The first requirement is met by enclosing the quote in (single or double quotation) marks. You can see an example of this below. The second requirement can be met in one of two ways. One method is to follow the quotation by a full description of the source, as shown below.

“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 First Principles of Interaction Design.[Online] http://www.asktog.com/basics/firstPrinciples.html [Accessed 14th September 2006])

This method works well when there are only a few references in the work and particularly when there are not multiple references to the same source.

For major projects it is usually a requirement to adopt the alternative, more formal, approach by providing a list of references to all the sources you have used for your work (usually at the end of the work). You then place references to this list within the body of your work. There are many “conventions” on how this should be done but all have the objective of linking something in the body of a report or paper to an entry in the reference list. One convention is outlined below.

If you were to write:

“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)

you could have the following entry in your list of references.


Please note that simply placing the source for your quote into your reference list is not, on its own, sufficient since it does not allow the reader of your work to judge exactly what has been drawn from the quoted source and what has not! That is the purpose of the first requirement (use of quotation marks around the quote).

If the cited work had two authors, one could reference this as below.
A useful language feature for programming in the large is the restriction of "name spaces" (Ghezzi and Jazayeri 1998).

The entry in the list of references would be as follows.


If, in your references, you have two entries for the same author and year you can resolve the ambiguity by labelling the years a, b, c etc, as below.

The computer, the interface, and the task environment all "belong" to the user, but user-autonomy doesn’t mean we abandon rules. The fundamental architecture of the system determines its efficiency, rather than the design of the interface. (Tognazzini 2003b).

Much further guidance on the details of referencing can be found on the web. A good UH source is http://www.studynet1.herts.ac.uk/ptl/common/LIS.nsf/lis/busharvard.

When you do attempt to make a direct quote you should ensure that you copy the quotation exactly!

General guidelines when directly quoting the work of others include:

- Use direct quotations only when the author's wording is necessary or particularly effective.
- Select carefully. A quotation should add weight to what you are trying to say, not simply repeat a point you have already made.
- Be sure to integrate all ideas into your own discussion. Introduce direct quotations with your own words. Explain their significance.
- Avoid quoting more than is needed.
- Use single and double quotation marks in a consistent manner.

2.2 USE OF PARAPHRASING

Instead of using a direct quotation from a published source a student might simply rephrase the material by changing a few words or rearranging their order. A rearrangement of the above example that could still leave the student open to a charge of plagiarism if unreferenced is given below.

The fundamental architecture of the system determines its efficiency, rather than the design of the interface.

This is an example of plagiarism because the intellectual input from the student only consists of a rearrangement of the word order and the substitution of a few synonyms but nothing more. Consequently, the original author should still be credited with this idea, as shown below. Note the lack of quotation marks since the quote is not an exact one, and the use of layout on the page to indicate the limit of the creditation.

The fundamental architecture of the system determines its efficiency, rather than the design of the interface. (Tognazzini 2003).

Caution is required when referencing other people’s work that you have paraphrased. Although no quotation marks are used to indicate the extent of the work being cited the student is still under an obligation to make clear what is/is not his/her own work. For example, below:

The computer, the interface, and the task environment all "belong" to the user, but user-autonomy doesn’t mean we abandon rules.
The fundamental architecture of the system determines its efficiency, rather than the design of the interface. (Tognazzini 2003).

the assumption would be that because of the layout of the text on the page only the second idea was being credited to Tognazzini. If, as is actually the case here, both sentences describe Tognazzini’s work we could write this as:

The computer, the interface, and the task environment all "belong" to the user, but user-autonomy doesn’t mean we abandon rules. (Tognazzini 2003).

The fundamental architecture of the system determines its efficiency, rather than the design of the interface. (Tognazzini. 2003)

Or, of course, combine the two ideas, as below.

The computer, the interface, and the task environment all "belong" to the user, but user-autonomy doesn’t mean we abandon rules. The fundamental architecture of the system determines its efficiency, rather than the design of the interface. (Tognazzini 2003)

2.2.1 On The Difficulties Of Paraphrasing The Work Of Others

It is, in fact, very hard to read a single piece of text and then rephrase it in one’s own words. However hard you try, the original words and phraseology will persist in your brain and be incorporated into your work leaving you open to a charge of plagiarism. One way to avoid this problem is to read several articles on the relevant topic. When you do this it all becomes “mixed up” in your brain, and what you then write will be your own words. The words may not be as “polished” as the original, but it will be your work.

If you don’t feel confident in your abilities to express ideas in your own words the make sure you properly reference the original source to avoid any accusation of plagiarism.

A good way to avoid plagiarism is to concentrate your efforts on writing about what you think and what you have done. If there is background material that needs to be presented then describe it in the context of your work and explain how it relates to your work If you find yourself wanting to cut and paste quantities of material from another source into your work then something is wrong!

2.3 REUSE OF COMPUTER CODE

Sometimes it is legitimate to reuse or adapt the computer code written by someone else. For example, you may want to open a socket connection using Java and have found some code to do this in a textbook. If you wish to reuse or adapt some code you should:

a) alert the reader that you have made use of someone else’s code by reporting it in the body of your report describing the source of the code (via the list of references, if used), the extent of its use and whether/how you have developed it (possibly putting some details in an appendix), and

b) indicate by means of comments in the code what has been written or modified by you (and how), and what has been used unchanged.

For example, were you to copy and adopt the Java method:

```java
public void listenSocket(){
    try{
```
server = new ServerSocket(4321);
} catch (IOException e) {
    System.out.println("Could not listen on port 4321");
    System.exit(-1);
}  

the code you report could be:

/* original code taken from http://java.sun.com/ commented out  
public void listenSocket(){
    try{
        server = new ServerSocket(4321);
    } catch (IOException e) { 
        System.out.println("Could not listen on port 4321");
        System.exit(-1);
    }  
} end of original code */

// start of adapted code
public void listenSocket(int port_no){
    try{
        server = new ServerSocket(port_no);
    } catch (IOException e) { 
        System.out.println("Could not listen on port :"+ port_no);
        System.exit(-1);
    } /* end of adapted code

where there is an entry in your references for http://java.sun.com/ giving the full URL you
have used
(http://java.sun.com/developer/onlineTraining/Programming/BasicJava2/socket .html in this case!)

Alternatively, with changes that are relatively small compared to the size of the copied code the
changes could be indicated by commenting out and replacing single lines of code.

A third alternative is to clearly mark the code with a pen, indicating what is, and what is not, your
work and what is copied.

What is not acceptable is to provide a large chunk of original text and another large chunk slightly
altered text and expect the marker to work out what you have and have not changed!

2.4 AUTOMATICALLY GENERATED COMPUTER CODE

Some Integrated Development Environments (IDEs) allow the user to specify a programming task in
high level terms and they will then generate the programming code for you. For example, one IDE
allows you to specify a database, an HTML table, and an SQL query and then creates the Java code to
connect to the database, apply the SQL statement, and then insert the results of that query into the
table.

Using these facilities of an IDE might or might not be an acceptable thing to do. If the specification
for the assignment explicitly says that you can use code-generation facilities then this is obviously not
a problem provided you comply with any restrictions or reporting provisions that are required.

If the coursework assignment says you shouldn’t use an automatic code generator, then don’t!

When explicit permission for using the code-generation facilities of an IDE is not given you should
seek the advice of your project supervisor or module leader. Caution is required here since with some
assignments a collection of technical tasks are set for the student to complete on the assumption that the code will be “hand written”. Use of code-generators in these circumstances prevents the markers from properly assessing your skills and knowledge and could result in a poor mark for the work.

In all cases if you have used a code generator you should:

a) alert the reader that you have made use of a code generator by reporting it in the body of your report, naming the code generator, describing how you used it and how you specified the required code to it (possibly putting some details in an appendix), and

b) indicate by means of comments in the programme code you submit what has been directly written by you, and what has been produced by the code generator.

2.5 DETECTING PLAGIARISM

You should be aware that we have several automated methods that make it easy to detect plagiarism; examples we use are:

- Turnitin for comparing reports with material on the Web and in databases from other institutions.
- The Ferret plagiarism detector for comparing students’ work with each other.

You should also be aware that the academic staff of the School are very experienced markers and are adept at being able to distinguish between students’ written work and that produced by professional writers such as academics, technical writers, journalists etc. If you can find some relevant work on the web to include in your assignment, so can we!

3 OTHER FORMS OF ACADEMIC MISCONDUCT

For completeness, please note that there are other forms of academic misconduct. These are described in UPR AS/C/6.1 (Appendix I).

Roger Oliver
Associate Head of the School of Computer Science (Academic)

With thanks to:
Ruth Barrett
Caroline Lyon
James Malcolm

October, 2006