Computerised marking of weekly mathematics assignments has several potential benefits: 1) randomised questions reduce plagiarism, 2) immediate feedback increases motivation and 3) staff-time spent on marking is reduced. We describe a practical way of introducing computerised marking into departmental teaching gradually, without disrupting current practice. The approach has been followed at the University of York during last year. It makes use of the open-source software packages AiM and Moodle.

The Challenge

It is reassuring to see that university mathematics courses follow the same pattern almost everywhere in the world: students attend two or three lectures every week and are then given a weekly assignment consisting of a collection of problems that the students take home and work on in their own time. They are usually given one week to complete the assignment and hand in their solutions. These are then marked, either by the lecturer or by a graduate assistant, and handed back to the student about a week later. The marks may contribute a small amount to the students’ final grade. Usually there is a regular problems class during which the assigned problems or similar ones are discussed in smaller groups.

This system, that is so universally followed, is designed to encourage the student to apply the knowledge acquired in the lectures and to make it their own. While the system is very sound in principle, there are some flaws in practice. Three of the major problems are:

1. Some students don’t actually attempt to solve the problems themselves but copy the solutions. Unfortunately these are usually the weaker students who are most in need of the practice afforded by doing the problems. Thus this contributes to the two-peaked distribution that we see in our classes, with a divide between the strong and the weak students.
2. Students have to wait about a week before finding out whether they solved the problem correctly. When they finally find out that there was a mistake, there is no incentive for them to correct it. The “learning from one’s mistakes” is not taking place.
3. The marking of the assignments takes up a lot of staff time that would be better spent in direct student contact.

Luckily all of these problems have an obvious solution:

1. Give every student a slightly different set of problems.
2. Give immediate feedback to the student.
3. Automate the marking.

Clearly this calls for a computer-based solution. The challenge is to implement this without disrupting the usual system that, after all, serves us quite well. We want to start to solve the problems while changing the existing system as little as possible and we want to allow the change to take place gradually.

The most important point to keep in mind is that the kind of problems that we like to assign to our students are those that the student needs to take home and
spend some time on to carefully work out the solution. We do not want to replace these by some quiz questions that the student answers while sitting at a computer. After all our goal is not to test the students’ knowledge but to motivate them to deepen their understanding by working on challenging problems. What we want is thus very different from traditional computer aided assessment.

The solution

I will now describe how we have met this challenge at the Department of Mathematics at the University of York. Our solution consists of providing the lecturer with the means to computerise some of the questions on the weekly problem sheets.

The technological heart of our system is AiM (www.aiminfo.net), a sophisticated CAA system that uses the power of the computer algebra system Maple (www.maplesoft.com) to randomise and to mark mathematics problems. AiM and its use has been the topic of previous articles in this publication [1] and elsewhere [2,3,4,5,6]. In this article I will concentrate on describing how we are employing it in our weekly assignments.

Students and lecturers interact with the system through Moodle (www.moodle.org), a particularly user-friendly and easy-to-use virtual learning environment. Most of our lecturers have been making course material available on the web for a long time. Moodle simply provides a convenient mechanism for doing this and for collecting everything in one place on the web. It also facilitates communication among the students and between students and lecturers.

Some details of how we went about the implementation will be described in the next section. In the following three subsections I will try to summarise the most important aspects of our approach using the same division into 3 points used in the introduction.

Randomised problem sheets

It has always been our practice to hand out typeset problem sheets that the lecturer has prepared using TeX or LaTeX. We will continue with this practice with only one subtle change: each student can receive a slightly different version of the problem sheet. The randomisation will be performed by AiM. Let us explain this with an example problem from vector calculus:

\[ I = \int_C x^3 y^2 dx + 2xy^3 dy \]

where \( C \) is the triangle with vertices \((0,a),(2,0),(-2,0)\).

In the past the lecturer used the following TeX source to typeset the problem:

\[
\text{Use Green’s Theorem to evaluate the line integral }
\]

\[
\int_C x^3 y^2 dx + 2 x y^3 dy
\]

\[
\text{where \$C\$ is the triangle with vertices }
\]

\[(0,a),(2,0),(-2,0)\].

To randomise this problem we have to use the following AiM source:

\[
\text{Use Green’s Theorem to evaluate the line integral }
\]

\[
\int_C \text{\@P@ } dx + \text{\@Q@ } dy
\]

\[
\text{where \$C\$ is the triangle with vertices }
\]

\[(0,a),(@xmax@,0),(@-xmax@,0)\].

You notice that the AiM syntax is a combination of Maple syntax and TeX syntax. The lines after the h> flag are Maple commands to choose the random parameters. The lines after the t> flag are TeX with an important extension: things enclosed in @...@ are evaluated by Maple and then substituted in. When processed by AiM this source will produce the same kind of problem as the one above but with randomised integrand and contour.

The system combines the randomised problems produced by AiM with the conventional problems to produce the required number of personalised weekly
problem sheets and sends them to the laser printer. The lecturer takes these to his lecture where each student picks up a random copy. Each random problem sheet has a number in the corner to identify it.

We like our students to discuss mathematics with each other. It has thus always been a pity that we had to discourage them from working on the assigned problems together. With the randomised sheets we can tell the students that they are allowed to discuss the solution method together as long as they afterwards apply it to their particular problem by themselves.

The differences between the problem sheets do not have to be large. They should just be large enough to discourage students to simply copy the answer from a fellow student.

**Immediate feedback**

Once students have worked out the solution to a computerised problem they can go to any computer with an Internet connection, browse to the course homepage on the departmental Moodle website and log in. In order to get the same version of the problem sheet that they worked on at home they type in the identification number from their problem sheet. They will then be presented with a webpage containing the questions and boxes into which they can type the answers.

---

**Fig 1 Webpage where students can enter answers and get immediate feedback**

All students have participated at an introductory session at the start of the year during which they were shown how to use Moodle and how to navigate to the on-line problems. Thus the individual lecturer does not have to explain these things.

After the students have entered a solution they click on “Mark” and immediately receive a reply stating whether the answer was correct or incorrect. This little bit of magic is again performed by AiM with the help of Maple. There are two ways for enabling AiM to do this:

1. If the problem has one correct answer then we can simply give this in the AiM source behind the `a>` flag. We can of course let Maple calculate the correct answer to the randomised problem. In the above example we would include the following in the AiM source:

   ```maple
   a>
   2*int(int(diff(Q,x)-diff(P,y),y=0..a*(1-x/xmax)),x=0..xmax)
   ```

   The student does not have to give the answer in any particular form because Maple can check for the equality between different expressions. So it does not matter in the above example whether the student types

   $$\frac{3}{10}a^4 - \frac{9}{10}a^3$$

   or

   $$(a-3)\times3\times a^3/10$$

   for example.

2. If the problem does not have a unique answer then one can tell AiM how to check that the student’s answer has the required properties. There is no limit to the sophistication in this marking procedure; the question author can employ the full power of Maple’s programming language.

The pedagogical benefit of the new system arises when students do not get the correct answer. They will then be encouraged to go away and find their mistake. When they have found the mistake they can come back to the computer and type in the corrected result. If necessary they can repeat this several times. There will be a penalty however for each wrong attempt in order to encourage students to do their solutions carefully the first time around. By default this penalty is 10% of the total mark, but this can be changed individually for each problem.

Note that students are given credit only if they finally arrive at the correct answer. This is a drastic change from current practice. It is conventional to give a student partial credit for a solution even if the answer is incorrect. Unfortunately this practice of giving partial credit for incorrect answers has led to a culture where some students just scribble something in the hope of attracting marks. These students are not being motivated to do the work carefully because they are given the wrong signal that getting the answer right is not so
essential. The computerised assignments on the other hand give the student both the possibility and the incentive to persevere until the answer is correct. These assignments thus train our students in the important skill of finding their own mistakes. We all know how important that is in our own mathematical practice.

Our experience with computerised assignments has shown two things:

1. The students do indeed persevere and by the time they get to the right answer they understand the material much better than after the first attempt.
2. Students are motivated to spend a lot more time on their assignments.

This second effect has to be taken into account when designing the assignments.

**Saving marking time**

Conventional marking of the weekly assignments takes up a lot of time that would be better employed differently. It is clear that having some computerised problems on a problem sheet cuts down on marking time because the marker no longer has to look at the solutions to these problems in detail. It is, however, important that this time saving is not lost again in the process of adding together the computer marks and the marks for the conventional problems. We solve this by letting Moodle automatically collect the computer marks. The marker simply types in the marks from the non-computerised problems and Moodle takes care of everything else. Both students and lecturers can always easily get an overview of the grades on the Moodle site.

We mark students’ work not only in order to give feedback to the students but also so that we can get an idea of how well our students are absorbing the material that we lecture on. Ideally our observations from marking feeds back into our teaching during lectures and problem classes where we can repeat or elaborate on points that have proven difficult in the assignments. Again computerised marking has several advantages. One is timing. We know how students are doing as soon as they start inputting their answers into the system ie. even before the due date. This gives us the ability to stress relevant points in the lectures before we have already moved on to other subjects. The second advantage is convenience. The AiM system provides the lecturers with several tools to analyse the results. The lecturer can pull up an overview of how the class did on a particular problem or look at all the attempts a particular student made. The lecturer can even ask for a list of all the students who made similar mistakes.

It is much better to go into a problems class after having seen an overview of the students’ performance, in particular as Moodle also shows this alongside the students’ photographs. Students in large classes are quite thrilled when they are approached in person with a verbal feedback on their recent work.

Clearly, besides the time saved through the automated marking, we also have to consider the time spent in setting up and testing the computerised problems. This is an important issue and will be discussed further in the next section.

**The implementation**

We developed and implemented the system in the summer of 2003 with the help of a small grant from the University of York through its Fund for Innovation and Development in Teaching and Learning.

**The technology**

The software required for the computerised assessment system can run on any modern PC under Windows or Linux with an internet connection. This computer must also have Maple version 7, 8, or 9 installed.

Students and lecturers access the system from their own computer through a web browser. They do not need any extra software installed on their computer.

As already mentioned, our system combines the VLE Moodle (moodle.org) with the CAA system AiM (aiminfo.net). One point worth stressing is that both Moodle and AiM are open-source software. This has allowed us to modify both in such a way that to the users they look like a single system. Most of the software development necessary to achieve this integration was done last summer by Alex Smith, then an undergraduate student in our Department.

The integration of the assessment system into a VLE has been crucial for its acceptance. The students immediately appreciated the convenience afforded by the VLE of having all their course materials and information available in one user-friendly place. This positive attitude towards the VLE they then also extended towards the assessment system that they saw as one piece of a package.

We decided to roll out the system department wide for all our undergraduate modules. Each module was given its page on the VLE and all students were enrolled automatically using data exported from the university’s student record system. It was made departmental
policy that all coursework marks for all modules were to be entered into this VLE. Every lecturer was then free to choose whether they wanted to use some computerised questions on their problem sheets or not.

**Authoring questions**

We knew from the experience at other institutions that the main stumbling block to the introduction of the AiM assessment system into a department is that lecturers do not have the time that needs to be invested initially to author the questions for AiM. In the summer we therefore assembled a team of seven undergraduate students whose task it was to transfer some of the existing assignment questions to AiM. Unfortunately the learning curve was steep and so between them the students spent in the order of 1000 hours to convert about 200 questions for 13 different undergraduate modules. If we had to repeat the exercise we would work with a smaller team.

What is really needed is a simple and convenient authoring tool that will allow lecturers to quickly computerise some of their simpler questions themselves. We have applied for a grant from JISC to develop such tools as well as for making mathematics assessment systems interoperable with each other and with other components of a managed learning environment. If this is approved then these tools will become available in spring 2005.

One important lesson that we have learned is that it is very important to test the marking routines very carefully. In the beginning it happened occasionally that a student's answer was marked as incorrect when in fact it was correct. This may also sometimes happen in hand-marking but it has a much more devastating effect when it happens in our system because it may lead students to waste a lot of time chasing for mistakes in their solution when there are none. Understandably this got students very annoyed. The solution is to have the questions tested thoroughly and independently by several marking assistants before their first use on students.

**Summary**

In this article I have pointed out some of the potential benefits of computerised assignments and have described a way of blending them seamlessly with the existing practice. The following points summarise some of the aspects that were important for the success of our approach:

- The computerised problems are integrated into the usual printed problem sheets. This encourages the student to take the computerised problems just as seriously as the conventional ones.
- The students enter their solutions through Moodle, which they use for their other course materials anyway. Moodle is accessed via the web, i.e., the student can do this from any computer.
- The students are introduced to the use of Moodle and AiM in a session at the start of term. Thus this does not steal any lecture time from the individual modules.
- The system is set up automatically for all undergraduate modules so that any interested lecturer is free to start using the system at any time.
- The marks for both computerised and conventionally marked problems are collected and combined automatically by Moodle. The students can check their marks and reassure themselves that they are correct.
- After the due date the students can download typeset solution sheets containing solutions to both the conventional problems and the randomised ones.

It is important to realise that only some questions lend themselves to computerisation. It is thus important that the system is designed in such a way that it can be used efficiently even on only a small proportion of the assignment questions in a module. It also helps to overcome initial scepticism if lecturers are given the possibility to start with just one or two computerised questions in their module in order to familiarise themselves with the concept.

The impact that the introduction of the new system has had on the experience of students and lecturers has been the object of a study by Richard Walker, the university’s learning technologist, and has been presented in [7].

This article has concentrated on presenting the use of AiM and Moodle in the weekly homework assignments. Of course the introduction of the software opens up several other opportunities which we will not discuss here.

We believe that any university department that is teaching some mathematics would profit from introducing a similar system. We are hoping to make our software available for this purpose with the help of a JISC grant. If this grant is approved we will also have the resources to help interested departments with the installation and initial rollout of the system. If you would like to be kept informed of future developments
you are encouraged to register at http://maths.york.ac.uk/serving_maths/. There is also a discussion forum where you can raise any questions that were not addressed in this article.

References