WeBWorK Tutorial

Michael Gage

March 2007

Contents

1 The Goal .......................... 2
   1.1 It increases the effectiveness of traditional homework ........ 2
   1.2 It increases the efficiency of traditional homework by: .... 2
   1.3 Key features: .................................................. 2

2 The WeBWorK roles: .............. 3
   2.1 Administrator: .................................................. 3
   2.2 Instructor: ....................................................... 3
   2.3 Student: .......................................................... 3
   2.4 Homework problem author ...................................... 3
   2.5 Tutorial plan .................................................... 3

3 Testimonials ................................... 4

4 Tutorial ................................. 5
   4.1 Login as a student: ............................................... 5
       4.1.1 Problem 1 - example of equation and numerical answers . 5
       4.1.2 Problem 2 - graph example ................................ 5
   4.2 Login as instructor ................................................ 6
       4.2.1 Problem 3 - adaptive equation answers .................... 6
       4.2.2 Problem 1 again – view code ................................ 6
       4.2.3 Problem 4 - answers with units ............................ 8
       4.2.4 Problem 5 - graph problems ................................. 8
       4.2.5 Problem 6 - interactive problem using JavaScript .......... 8
       4.2.6 Problem 7 - interactive problem using LiteApplet ....... 8
       4.2.7 Problem 9 - uses a java applet to display 3D images ...... 8
       4.2.8 Problem 8 - a problem we’ll delete ....................... 10
       4.2.9 Add a student to the course .................................. 11
       4.2.10 Change a student to an instructor ......................... 11
The Goal

Make homework more effective and efficient and promote active learning by students.

1.1 It increases the effectiveness of traditional homework

1. Providing students with immediate feedback on the validity of their answers and giving students the opportunity to correct mistakes while they are still thinking about the problem. As one student said, “I can fix my mistakes while [the] problem is fresh in my mind.”

2. Providing students with individualized versions of problems which means that instructors can encourage students to work together; yet each student must develop an answer to his or her own version of the problem.

1.2 It increases the efficiency of traditional homework by:

1. Providing automatic grading of assignments.

2. Providing information on the performance of individual students and the course (or section or recitation) as a whole.

1.3 Key features:

1. Using WeBWorK, instructors can ask most questions typically found in mathematics and other scientific textbooks as well as more advanced interactive questions.

2. Students persist with WeBWorK. At Rochester we find almost all students complete each homework set until most problems are correct.
2 The WeBWorK roles:

2.1 Administrator:
- creates courses

2.2 Instructor:
manages courses
- Enrolls students in the course
- Creates homework sets and assigns them to students
- May view student progress in answering questions

2.3 Student:
- Answers questions in the homework sets
- Answers for each problem are automatically checked and the grade recorded.
- Usually students are allowed to correct any mistakes in their homework problem – the instructor can impose a limit on how many times a homework problem can be corrected.
- At Rochester we have been using the model of homework, where feedback and correction However, this “gateway quiz” model is also available in a “beta version” in WeBWorK and is being used at the University of Michigan and several other institutions.

2.4 Homework problem author
The instructors may create their own problems or they may choose problems from the problem library. There is already a large collection of problems for calculus, pre-calculus, linear algebra, differential equations and some probability and statistics.

We plan to create and curate an even wider selection of problems for the MAA digital library site.

2.5 Tutorial plan
We will practice using WeBWorK as a student and then add homework sets to a course as an instructor.

If there is time we will look briefly at how new homework problems are created.
3 Testimonials

The educational advantages of immediate feedback are emphasised repeatedly by both instructors and students

- “I can fix my mistakes while the problem is fresh in my mind.”
- “WebWork is the only way that I can push myself to really do something about the homework.”
- “I don’t have to wait for lecture to see if I’m doing it right.”
- “It makes you want to redo it; after finding an answer, you feel accomplished, immediate feedback makes sure you have accomplished something.”
- “Our analysis shows that nearly all Rochester students complete each homework set until most problems are correct.”
- “The Calculus classes went very well. I cant get over how important the e-mail the instructor button is.”
- A Rutgers study found the “correlation between attempts and percentage of problems solved was a remarkable .944 suggesting that once students began a problem they persisted until they had solved it” (Hirsch and Weibel, 2003)
- A three-part study at Virginia and Rochester reveals that WeBWorK “spawned noticeable changes in persistence and determination when it came to working through assignments” (Davison, 2004).

Testimonials about WeBWorK from instructors:

WeBWorK seems to cause students to work harder on their homework and it also encourages them to work together more closely than they would otherwise. It is most gratifying to walk through our study center and see groups of eager and animated students discussing the meaning of variables, equations, functions, derivatives, and integrals. Since WeBWorK has an easy and e mechanism it also encourages and intensifies interaction of the students with their instructor.

Peter Alfeld, University of Utah

It is so valuable to center a mathematics course around students solving problems, instead of around students watching the instructor solve problems. One learns math only by doing it. Without instant grading of math homework, even if students do math homework assignments, they do not know if they have done them correctly. Being told immediately that their answer is wrong is a strong motivation for students to keep working on a problem. Engaging students to stick with a problem until they get it right is an extremely powerful strategy for good learning. The Rutgers study of WeBWorKs impact demonstrated this claim in spades.

Alan Tucker, SUNY-Stony Brook
The WebWorK project not only provides an extremely useful pedagogical tool, but as an open source project is not bound to one textbook or publisher, and so embodies the flexibility to be tailored to each adopters needs. No commercial product can o products which have similar capabilities, but they inevitably pale by comparison. Tom Shemanske, Dartmouth

[We] have found WebWorK to be a valuable support for courses in our undergraduate mathematics program. The response to the introduction of the homework is almost universally positive, both from students and faculty, and we will be using the homework in all of Calculus I this winter, serving another 2500 students per year. Once this is proven successful we will almost certainly extend its use to our entire introductory program, which enrolls over seven thousand students a year. In short, WebWorK is likely in the next few years to have a significant impact on the education of almost all students who go through a standard mathematics course [here]. Robert Megginson and P. Gavin LaRose, University of Michigan

There are more instructor comments at http://webwork.rochester.edu/webwork-summary.pdf

4 Tutorial

4.1 Login as a student:

Use student1 as login name and student1 as password

4.1.1 Problem 1 - example of equation and numerical answers

Things to notice:

- When you enter an equation your entry is typeset. Enter \((x + 1)^3/10x\) in the first entry. Notice the denominator might not be what you expect
- Enter \((x + 1) * 5\) – notice that the mismatched parentheses are highlighted
- Enter the correct answer as \(ax + b\) and also \(a(x + b/a)\) - both are accepted.
  - Each of you has a different \(a\) and \(b\) because of the randomizing behavior.
- Enter the numerical answers: notice that the instructor can require the student to do the calculation or allow webwork to do the numerical calculation for them.

4.1.2 Problem 2 - graph example

Things to notice:

- Click on the graph for a larger view.
- In this example WebWorK indicates which part of the answer is wrong – this feature can be turned on or off by the instructor.
• The problem grade is calculated every time you do a problem. The highest grade is retained.

![Figure 1: Logout](image)

### 4.2 Login as instructor

First click logout in the upper left corner and confirm.
Then use **profa** as login and **profa** as password to login again.
As an instructor you will have more options.

#### 4.2.1 Problem 3 - adaptive equation answers

Things to notice:

- Enter \( \cos(x^3) \) or \( \frac{\cos(x^3)}{3} \) + 1 or \( \frac{\cos(x^3)}{3} + \sin^2 x + \cos^2 x \) – each will be marked correct.

- WebWorK uses numerical sampling – so two functions are equivalent if they agree on 5 random points. The number and domain of the points chosen can be adjusted. This criteria works surprisingly well.

#### 4.2.2 Problem 1 again – view code

- Click on problem 1 lower down in the left margin (or navigate to problem 1)
- Click on the link “Edit this problem” to open an editor for the code in a new window.
- We probably won’t have time to practice tweaking problems, however note the links at the top for obtaining various types of documentation for help.
Problem 1

Editing set Introduction/problem 1 in file [TMPL]/introduction/s2_2_1.pg

1. Manuals 1. macro list 1. pod docs 1. report problem bugs

```
/#DESCRIPTION
## Find a derivative of a polynomial, evaluate it at a point
## ENDRDESCRIPTION

DOCUMENT();
# This should be the first executable line in the problem.
loadMacros( 
    "PGbasicmacros.pl",
    "PGchoicemacros.pl",
    "PGauxiliaryFunctions.pl"
);

TEXT(beginproblem);
#showPartialCorrectAnswers = 1;
$s1 = random(2,7,1);
$s2 = random(2,12,1);
$s3 = random(1,40,1);
$s4 = random(1,9,1);
$s5 = random(1,5,1);

# View using seed 4040 and display mode images

Add to set Introduction as problem

Save to [TMPL]/introduction/s2_2_1.pg and View

Create a copy at [TMPL]/introduction/s2_2_1.pg and use in set Introduction/problem 1

Select above then: Take Action! in another window
```

Figure 3: PG editor
4.2.3 Problem 4 - answers with units

Close the PG editor window to return to problem 1. Then click on the link in the left margin (or otherwise navigate to problem 4).

Things to notice:

• Enter the answer with units: \( \sqrt{x^2 + b^2} \) m
• Now enter it in centimeters: \( \sqrt{x^2 + b^2} \times 100 \) cm
• Try some other correct and incorrect answers

4.2.4 Problem 5 - graph problems

• The graphic picture is generated “on the fly” as you request the problem. It is slightly different for each student.
• This problem is a bit more difficult than you might expect
• The instant feedback keeps the student at the task until they get it.
• Notice that in this case WeBWorK does not indicate which answer is wrong.
• As an instructor you can click the “show correct answer” checkbox to see what WeBWorK thinks is the correct answer.

4.2.5 Problem 6 - interactive problem using JavaScript

• The student MUST use the Newton quotient to calculate the derivative \( f'(a) \) since there is no formula.
• Enter points \( x \) near \( a \) and click the \( - - f - \rightarrow \) button to calculate \( f(x) \)
• Try some other correct and incorrect answers

4.2.6 Problem 7 - interactive problem using LiteApplet

(java applet) by Frank Wattenberg

• Move the pointer to obtain \((x,y)\) coordinates
• Notice that in order to find the distance the student must also measure the legend on the map in order to calibrate the coordinates.

4.2.7 Problem 9 - uses a java applet to display 3D images

(created by Davide Cervone)
Figure 4: Hmwk Editor
4.2.8 Problem 8 - a problem we’ll delete
from the homework set

- Click on Instructor Tools/Hmwk Sets Editor/Introduction in the left margin.
- Find problem number 8 in the list and check the “delete” checkbox
- At the bottom of the page click the “force problems to be numbered consecutively” checkbox
- Click the save button. Notice that there are now only 8 problems.
- Change the due date and and the answer date at the top of the page. You can copy and paste the format
- Click the save button.
- Notice that the timezone is still Eastern Daylight Time (EDT). We’ll change that
- Click on the “Course configuration” link in the left margin
- Enter “Asia/Muscat” as the new timezone and click save.
- Finally click the “Homework Sets/ introduction” link in the top left to return to the main homework page.
- Notice that the time zones have changed.

Figure 5: Course Configuration Editor
4.2.9 Add a student to the course.

- Click “Classlist editor” in the left margin
- Click “add \(< n \) student(s)” – you can change the number of students to add to 2
- Click “Take action” button
- Fill in the student data. The student’s initial password is the same as the studentID.
- Select the sets to be assigned to the student.
- Click “Add Students” AT THE BOTTOM of the page.

4.2.10 Change a student to an instructor

- Click on the pencil next to future instructor (for example student2 at the bottom of the page).
- In the box all the way to the right change the permission level from 0 to 10
- Click the “Save changes” button
- Click the “Take action” button
• If only some students are shown choose the “Show all users” popup in the first line and then click “Take Action”

4.2.11 Add many students

• Click “Classlist editor” in the left margin

• Click “Import users from file demoCourse.lst ...”

• Notice that several “guest” or “practice” users have been added. These users’ grades are not permanently recorded. Students can use these accounts for practice if they are available.

• Notice that these “practice” users have not been assigned any sets (the column reads “0 out of 1”),

• You can assign a set to an individual student by clicking on the link in the “Assigned Sets” column.

![Main Menu](Image)

<table>
<thead>
<tr>
<th>Select</th>
<th>Edit Set Data</th>
<th>Edit Problems</th>
<th>Edit Assigned Users</th>
<th>Visible</th>
<th>Open Date</th>
<th>Due Date</th>
<th>Answer Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>10</td>
<td>10/18</td>
<td>Yes</td>
<td>02/25/2007 at 03:07pm GST</td>
<td>06/02/2007 at 10:07am GST</td>
<td>06/02/2007 at 10:07am GST</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7: Hmwk Sets Editor

4.2.12 Import a homework set

This imports a saved or precreated homework set from a set definition file (.def file).

• Click “Hmwk set editor” in the left margin
• Click “Import a single set from ...” and change “following file(s)” to setOrientation.def
• Choose “assigning this set to all current users”. You can leave the “set name” blank and the set will be called Orientation.
• Click “Take action” button

4.2.13 Assign the “Introduction” set to the practice users
• Click in the “Edit Assigned Users” column
• On the next page click on “Assign to All Current Users”
• Notice that this set has now been assigned to the practice users.
• Return to the Hmwk Sets Editor (click the link in the left margin).

4.2.14 Create a new empty homework set
Finally we can compose our own homework set from the library. First create an empty homework set.
• Click “Hmwk set editor” in the left margin if you are not on the homework editor page.
• Click “Create a new set named < mySet > as new empty set”. (You can use your own name for “mySet”.)
• Click “Take action” button
• Assign the set to yourself – (Click in the “Edit Assigned Users”, check profa, and click “Save” at the bottom of the page.)
• Click “Library Browser” link in the left margin

4.2.15 Use library browser to add problems
to homework set
Choose problems from the library
• First choose “mySet” from the top popup button to be the “target” set for adding problems.
• Click “Problem Library”
• Select a problem collection (Subject/Chapter/Section)
• Click “View Problems” to see the problems in the collection
• Scroll down and select the problems to be added to the target homework set.
4.3 Other ideas for practice

- Practice by adding the homework sets “Knoxville MAA 2006 cervone” and “2007 new orleans gage”
- Browse through the presentations by Davide Cervone made last summer and Mike Gage made in January.
- Add the homework set “set0” and practice editing the text for these problems. It’s a good idea to make only very slight changes and then view the result, particularly while you are experimenting. You can revert back to the original problem as long as you only view the problem and don’t “Save as”.
- I can restore the course so don’t worry about damaging it permanently by a mistake – and don’t count on any changes you’ve made lasting for a long period of time.
- Practice adding more problems from the “Problem library” using the Library Browser. There are also older libraries created by each university – most but not all of these problems are now in the national “Problem library”. The Union College problems are among the newest. They are good examples.
- While viewing problem 1 (or any problem with equations) choose the different display options in the lower left hand corner to see three different
methods for displaying typeset mathematics. We have been using “images” mode.

5 Timeline

• Fall 1996 - WeBWorK first used in classes at U of R
• Spring 1999 - NSF support for WeBWorK at U of R
• Fall 1999 - WeBWorK received ICTCM award for Excellence and Innovation with the Use of Technology in Collegiate Mathematics
• Spring 2003 - WeBWorK 2 is first released
• Summer 2004 - MSRI sponsors WeBWorK programming workshop
• 2007 - More than 80 universities and colleges serving WeBWorK and more than 100 courses hosted on U of R server
• August 2007 - American Institute of Mathematics sponsors workshop on WeBWorK development and outreach

6 WeBWorK Team

In addition to the original WeBWorK developers Prof. Michael Gage and Prof. Arnold Pizer, the evaluator Dean Vicki Roth and the chief programmer Sam Hathaway there are many, many people have contributed and continue to contribute to the development of WeBWorK.

A partial list is at http://math.webwork.rochester.edu/docs/credits.html

Please join us. WeBWorK is open source: http://webwork.maa.org