

**Maths-CAA Series  
October 2003**

**Anatomy of a Mathwright Problem Object**

**Contents**

<b>Introduction</b>	2
<b>Current State: SAT Math Practice and Tutoring Center 1</b>	4
<b>Planned Enhancements</b>	12

# Anatomy of a Mathwright Problem Object

James E. White

Director, New Mathwright Library and Cafe

Email: mathwrig@gte.net

**Abstract:** While multiple-choice tests are generally poor measures of mathematical understanding, multiple-choice problems can, in a dynamic and interactive setting, actually improve a student's grasp of a mathematical topic. This is, of course, a difficult thesis to defend, and the aim of this paper is to illustrate, rather than to prove it. For that, the article has an interactive part, where you can see for yourself how *problem objects* may be put to use in a task of formative assessment.

The focus of the paper, however, will be a discussion of the structure of the "problem object" itself, how it is presently used in Mathwright, and how it can be extended to support the creation of new interactive settings in which students can sharpen their own skills.

## 1. Introduction

The interactive setting that I will use for illustration in this paper is a *Mathwright Microworld* (see the MSOR Connections article: *Beyond Applets*, Nov 2001, Vol 1, number 4, or the *Journal of Online Mathematics and its Applications* article: *Introducing Mathwright Microworlds*, Vol 2, 2002). The Microworld is entitled "SAT Math Practice and Tutoring Center 1". The SAT is the Scholastic Aptitude Test, and many American Colleges and Universities use its results in the process of deciding whether to admit (or to offer scholarships to) applicants.

This Microworld uses 25 "problem objects" and it generates an essentially unlimited number of multiple-choice tests that students may use to practice for the Mathematics section of the SAT. Students may read the Microworld online in their browsers in "test mode," in which case they are given 30 minutes to complete a 25-question test. At the end of that time, or whenever they complete the test (if sooner), they are taken to a "review page" that displays their overall score, the time used, and that reports which questions they answered correctly, and which they did not. At this point, the student may do one of two things. She may choose to enter "answer" mode, in which she may visit any problems to see the correct answers to each one.

Up to this point, the interaction is a fairly standard example of computer-aided instruction. But the student may also enter "practice mode." In this mode, she may return to any problem and practice it. Each time she selects "New Prac-

tice Problem" the system generates a new multiple-choice problem of the same type as the one on the test. This time, she receives immediate feedback when she chooses her answer. She is told whether her answer is correct or not. The student may practice a problem as often as she likes, or she may go on to practice other problems from the test. When the student is satisfied, she may return to the review page, and will find a graphical record of the ratio of correct answers to incorrect answers for each problem she practiced, along with her original test score.

The advantage of a strategy like this is obvious. The student can choose what to practice, and how often to practice it. At the end of the session, or if she returns to the Microworld at a later time, she will be given an entirely new timed test, based on the same 25 problem objects. So she may judge for herself whether her performance improves over the course of time.

The title of the Microworld: "SAT Math Practice and Tutoring Center 1" hints that there will be a sequel. The sequel will in fact provide an additional service that will truly justify this use of problem object to support interactive multiple-choice environments. This additional facility is the following. In practice mode, students will be able to ask for a step-by-step solution for each generated problem. The explanation will use the parameters of the problem itself, so that the answer will be specific to it. This will explain to them how they should approach future problems of that type. Several interactive books at the Mathwright Library, (for example, the free *Exploring Quadratic Functions* at the MATH Cafe, which you will be able to read in your browser after you download the Player for this article) utilize this procedure, and have been used successfully in distance learning courses.

The Mathwright Player that you will need to experiment with "SAT Math Practice and Tutoring Center 1" in your browser is a free one-time download, and you will see below what to do to get it. Once you have the Player on your machine, you may read any of the 12 complimentary Microworlds at the MATH Cafe in your browser. If you join the library, you will have access to all 230 of our Microworlds and WorkBooks, either online or offline.

If you would like to visit the Mathwright Microworld, *SAT Math Practice and Tutoring Center 1* now, then please follow these steps. You must be on a Windows platform (Windows 98/ME/NT/2000 or XP) and you must have an ActiveX-enabled browser such as Internet Explorer 5.0 or later. To read the Microworld, please click the following link to get the Player, and then start reading the Microworld. You will be able to use the same Player to read other Microworlds at the Library at the MATH Cafe. If you have already downloaded the Player,

then simply visit the Microworld.

If you would like to continue to read about “problem objects” and their architecture, then please read on. Later, you may return here to get the Player and view the Microworld. In what follows, I wish to discuss the structure of the problem object as it is currently implemented in Mathwright, and as it will be extended in the future.

### Current State: SAT Math Practice and Tutoring Center 1

The base language for Mathwright is called **Mathscript**, which is written in LISP on top of Java. Being LISP based, Mathscript is a flexible and dynamic scripting language that allows authors to create two types of “objects.” The first of these are **screen objects**, such as 2D and 3D graphing windows, Mathematical Expression windows, textfields, buttons, checkboxes, and so on. And the second of these are **mathematical objects** such as functions, mathematical expressions, matrices, vectors, programs, numbers (exact rational, decimal, unlimited precision, complex, quaternionic), and strings, lists, and commands. These objects are united in a multi-page microworld to form a constellation of interacting entities, by means of **scripts** that the author writes and saves as files, or attaches to screen objects, pages, and the Microworld itself.

The scripting language, Mathscript, resembles Pascal in many ways. But it is also object-oriented, allowing the author to create for a Microworld, her own hierarchy of classes of special-purpose objects (object types) and then to allow scripts to create these objects on the fly in response to user input. The problem object class, which is the support of *SAT Math Practice and Tutoring Center 1*, is an example of such a user-defined class. Problem objects send and receive messages, utilize methods to perform actions, and they have a list of properties (or attributes) that define their state. The Microworld creates 25 problem objects at startup, one for each problem in the “test mode” SAT Exam.

Each problem object has certain visual attributes and certain (hidden) data attributes. Visually, a problem object appears on a page. An example of such an object is shown below. The red comments are not part of the problem object, but are included here to allow us to describe this object.

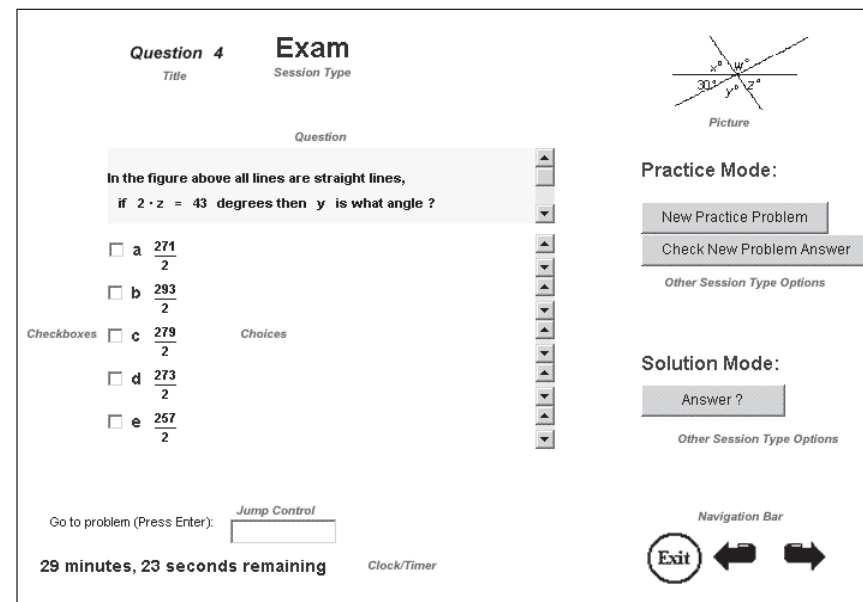


Figure 1:

This problem object: **Question 4** was created when the Microworld was opened, but its data and some visual attributes are created each time the object is “activated”. This may happen when the reader comes to the page for the first time in test mode. If the reader was in practice mode, each time she pressed “New Practice Problem” a new set of data and visual attributes would be created for the newly activated Question 4. In test mode or solution mode, however, the object is activated only once, so that if a reader returns to the page during the test or while reviewing it, she sees the same problem.

For example, the Question 4 shown below was given different data and visual attributes when the reader came to the page after asking for a new test. When a new test is requested, each problem object is reactivated so that the reader sees an entirely new test. I shall discuss the activation process in some detail below.

The visual attributes of a problem object like Question 4 that do not change when the object is activated are the *Picture* in the upper right hand corner, the *Title* (Question 4), the list of *Checkboxes* on the left, the *Navigation Bar* and *Jump Control*. The *Clock/Timer* counts down the time and is a part of every

problem object when the *Session Type* is **Exam**. It is not visible when the Session Type is **Practice** or **Solution**. The visual attributes of Question 4 that change with each activation are the *Question* and the *Choices*.

The data attributes of a problem object like Question 4 change each time the object is activated. This is what makes it possible to generate new versions of any problem object on the fly. The crucial data item is of course the *Question*. The question has two parts: text that does not change, and parameters that are generated at each activation of the object. For example, for Question 4, the static text is:

In the figure above all lines are straight lines, /n if /p = /p degrees then /p is what angle?

The escape characters: /v and /p represent parameters that will be supplied when the problem is generated. The escape character /n simply means to skip a line. If the parameter is represented by /v, it means that the parameter will be an integer, a decimal number, a vector, matrix, or a string. If the parameter is /p, it means the parameter will be an exact fraction (rational number) an exact rational algebraic expression (printed with raised exponents, numerators over denominators, etc.) or perhaps a complex number. In this problem, the  $2z$  is an algebraic expression, as is the  $y$ . In general, for this question, one may randomly be required to solve for  $x$ ,  $y$ ,  $z$ , or  $w$ , and one of the other variables will appear in the equation.

When the question parameters are selected, then five distinct *Choices* are generated as algebraic expressions, integers or fractions, or possibly vectors, matrices or strings, and they are assigned to the *Checkboxes*. The correct choice (e in this case) is noted by the system. On each activation, the correct answer is of course randomly assigned to a checkbox. Each checkbox has the property that when checked, it unchecks all the other checkboxes.

When the reader leaves the page (in test mode) her selection is noted and added to the tally. Finally, if time runs out, then the reader is taken to the review page to see her final score.

Below is a new activation instance of Question 4, generated when the exam was retaken. It is extremely unlikely that any reader will ever see two identical activation instances of Question 4.

**Question 4 Exam**

In the figure above all lines are straight lines,  
if  $2 \cdot x = 113$  degrees then  $w$  is what angle ?

a  $\frac{187}{2}$

b  $\frac{219}{2}$

c  $\frac{225}{2}$

d  $\frac{217}{2}$

e  $\frac{179}{2}$

Practice Mode:  
New Practice Problem  
Check New Problem Answer

Solution Mode:  
Answer ?

Go to problem (Press Enter):

21 minutes, 35 seconds remaining

Exit

Figure 2:

The frameworks for the problem objects are actually created, one at a time, in a separate Microworld called the **Problem Designer**. The Activation Rules for these problem objects are then created in scripts in the Microworld that shows them. Each **Page Script** (which is run when the page is entered) contains a program that activates the object just once for each test. Thus, these frameworks do not actually provide instantiated objects or their final data. They simply provide a template that the activation code can then use to provide new data. For example the framework for object **Question 4** was created as below in the Designer and then saved to disk.

**Problem Designer**

Restore Save

Problem Name: prob4 Practice Name: prac4 Wallpaper File: prob4.gif

**Problem Text:**  
 In the figure above all lines are straight lines, if  $\angle p = \angle q$  degrees then  $\angle r$  is what angle?  
You may place parentheses preceded by 3 backslashes, i.e. \\( in the yellow fields.

**Arguments for the Problem Text (one for each /p or /q, arranged in order):**

y 55 x

Choices for Answers:	Output Type:	Arguments for MathEdit /v and /p:	Correct Answer:
<input type="checkbox"/> a	Text	102	<input type="checkbox"/> a
<input type="checkbox"/> b	Text	95	<input checked="" type="checkbox"/> b
<input type="checkbox"/> c	Text	93	<input type="checkbox"/> c
<input type="checkbox"/> d	Text	92	<input type="checkbox"/> d
<input type="checkbox"/> e	Text	96	<input type="checkbox"/> e

Figure 3:

Now of course, the activation scripts that actually initialize the problem objects are the main item of interest. They vary from object to object, depending on its requirements, but they all do a number of things in common. Among these are:

- Creating a permutation object to rearrange the arguments for the problem text
- Creating the arguments for the problem text itself if numeric or algebraic expressions need to be generated
- Automatically reducing all generated fractions to lowest terms (if required) and simplifying generated algebraic expressions
- Calculating the answer based on these preliminaries
- Calculating four *distinct* plausible but incorrect answers
- Creating a permutation object to assign these answers as choices to checkboxes

These scripts are written in Mathscript, and so they have access to exact rational symbolic calculations, as well as the application of expert system rules for simplification. I will illustrate a few of the tasks below with five different problem objects.

## Example 1:

**Question 1 Practice**

Which of the following is less than  $\frac{3}{2}$ ?

a  $\frac{13}{8}$

b  $\frac{11}{6}$

c  $\frac{8}{5}$

d  $\frac{5}{3}$

e  $\frac{7}{6}$

The question generated is: “Which of the following is less than  $\frac{3}{2}$ ?” The system first generates a positive rational number,  $\frac{3}{2}$ , making sure it is reduced to lowest terms. Next, it generates four incorrect answers by randomly creating four distinct positive rational numbers, adding each one to  $\frac{3}{2}$ . In each case, it reduces the result to lowest terms. Next, it randomly generates rational numbers and subtracts them from  $\frac{3}{2}$  until it gets a positive result (in this case,  $\frac{7}{6}$ ). This it stores as the correct answer. This example illustrates the simple dependence on rational arithmetic.

## Example 2:

**Question 2 Practice**

If  $x = 1$ , what is the value of  $3 \cdot x^2 + 9 \cdot x + 12$ ?

a 6

b 66

c 24

d 132

e 12

The question generated is: "If  $x = 1$ , what is the value of  $3x^2 + 9x + 12$ ?" Here, there are two problem parameters: the value of  $x$  and a quadratic expression in  $x$  to evaluate. The system generates an integer value of  $x$  ( $x = 1$ ) and creates a random quadratic function with integer coefficients. Next, it generates four distinct incorrect answers and offers the list of choices above. This example illustrates the generation of algebraic expressions for the problem text.

### Example 3:

**Question 11 Practice**

Which of the 5 following equations defines the parabola on the right?

a  $x = 2 \cdot y^2 - 4$   
 b  $x = 2 \cdot y^2 + 3 \cdot y - 3$   
 c  $x = 2 \cdot y^2 + y - 4$   
 d  $x = 2 \cdot y^2 + 2 \cdot y - 3$   
 e  $x = 2 \cdot y^2 - y - 3$

**Practice Mode:**

New Practice Problem

Check New Problem Answer

The question generated is: "Which of the 5 following equations defines the parabola on the right?" This problem object creates an equation in variables  $x$  and  $y$ . It also contains a *window* in which it draws the solution curve for that equation. Next it generates four incorrect equations and permutes them to create the choices, which are algebraic expressions. Like all problem objects, it creates a new question and curve for each new activation. In its present form, a problem object can have at most one *window* (either 2 Dimensional or 3 Dimensional) for the display of its problem. This example illustrates the generation of graphics for each problem activation.

### Example 4:

**Question 8 Practice**

If  $r$  is a multiple of both 24 and 10, which of the following must be a factor of  $r$ ?

a 125  
 b 120  
 c 119  
 d 126  
 e 122

The question generated is: "If  $r$  is a multiple of both 24 and 10, which of the following must be a factor of  $r$ ?" Here, the system generates two random positive integers and then calculates their least common multiple. In this case, 120 is the answer. Here, we illustrate integer divisibility and factorization operations.

### Example 5:

**Question 9 Practice**

If the following is simplified, then

$$\frac{2 \cdot x^2 + 24 \cdot x \cdot y - 26 \cdot y^2 + 3 \cdot x \cdot z + 11 \cdot y \cdot z + z^2}{x + 13 \cdot y + z} \text{ is ?}$$

a  $2 \cdot x + z$   
 b  $2 \cdot x - 2 \cdot y$   
 c  $2 \cdot x - 2 \cdot y + z$   
 d  $2 \cdot x - 3 \cdot y + z$   
 e  $2 \cdot x - 5 \cdot y + z$

This question requires the script to generate two linear trinomials in variables  $x$ ,  $y$ , and  $z$  and to multiply them. It must then simplify the product as quadratic expression in these variables to get the numerator of the rational expression. Now the choices are also algebraic expressions: trinomials again. So the script implements an algorithm that compares them to guarantee that they are distinct before placing them in the list. This example illustrates the use of exact symbolic comparisons.

### Planned Enhancements

The real pedagogic promise for problem objects is the possibility of creating tree-like taxonomies of problem types that can guide a student through a series of increasingly sophisticated problems. There are many examples of strategies that require prior understandings and skills in order to be properly assimilated. These strategies can be organized and illustrated in a tree of problem objects that are chosen in advance.

Approaches such as this have of course been in use for decades. They are a common feature of CAI methods. Problem objects can, however, offer something new. It is possible to equip problem objects with the capability of providing step-by-step explanations of how each solution was obtained. Of course one may always give abstract explanations, but it is also feasible to provide such explanations using the generated parameters of each activation of a problem object. In addition to this, problem objects may allow the student to vary parameters herself and see the step-by-step solution to her variation of the problem.

This idea, which has guided the overall development of Mathwright, has already been implemented in several Microworlds by Mathwright authors. It is inspired by Jean Piaget's observation that the learner only ever acquires new understanding by asking (and answering) her own questions. As long as computational environments ask the questions for the learner, then tell her whether her answer was correct or not, they cannot teach. A teacher must listen for a student's question, and, at that moment, give a meaningful response.

The Microworld: *Exploring Quadratic Functions* by Dr. Samad Mortabit at the MATH Café in the Mathwright Library uses this strategy. You may view it in your browser once you download the Player for this article. The author provides a button "get an equation" that generates a random quadratic equation (in this case,  $x^2 - x - 2 = 0$ ). When the user pushes "graph parabola" it draws the graph, resizing the screen and highlighting the zeros, if any. Next, when the reader presses "solve equation" the system gives a step-by-step solution, as

indicated below on the left. Since the entire solution does not fit in the picture, it is included as text below this picture.

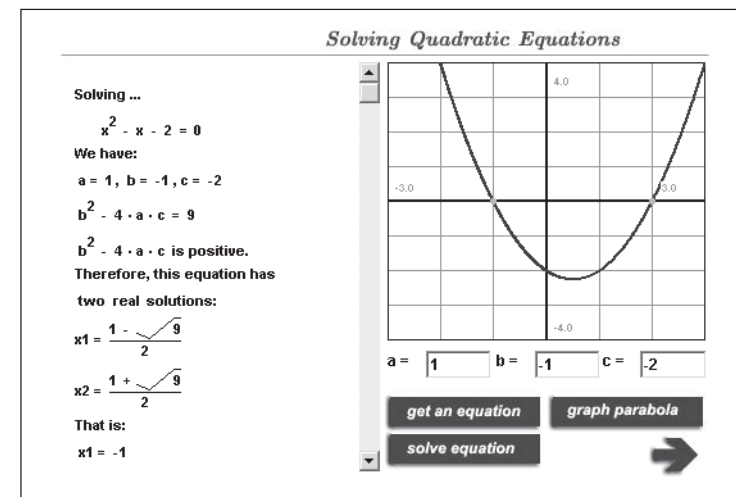


Figure 4:

Solving...  
 $x^2 - x - 2 = 0$   
 We have:  $a = 1, b = -1, c = -2$   
 $b^2 - 4 \cdot a \cdot c = 9$   
 $b^2 - 4 \cdot a \cdot c$  is positive.  
 Therefore, this equation has two real solutions:  
 $x_1 = (1 - \sqrt{9})/2$   
 $x_2 = (1 + \sqrt{9})/2$   
 That is:  
 $x_1 = -1$   
 $x_2 = 2$   
 or approximately:  
 $x_1 = -1$   
 $x_2 = 2$

At this point, the student might ask what happens if the coefficient  $a$  is negative. She may then simply type values for the coefficients:  $a$ ,  $b$ , and  $c$  in the boxes. For example, if she chose  $a = -1$ ,  $b = -4$ , and  $c = 3$  and pressed "graph parabola" the downward-turning parabola you see below is drawn. And if she then asks "solve equation" the system generates the solution. The entire text of the system response appears below the picture.

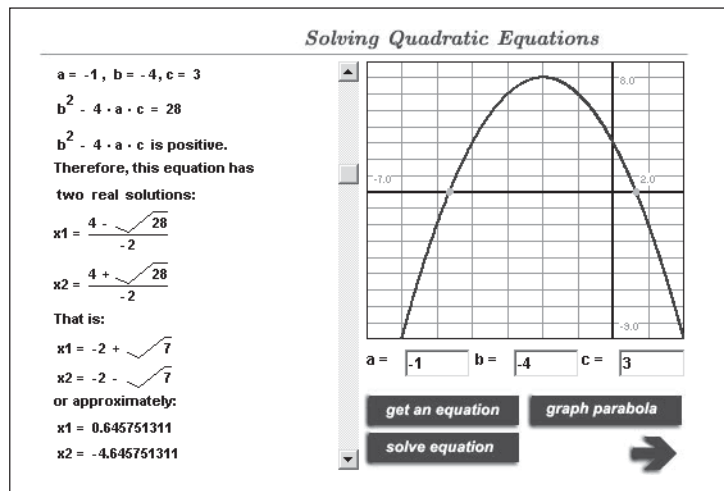


Figure 5:

Solving...

$$-x^2 - 4x + 3 = 0$$

We have:  $a = -1$ ,  $b = -4$ ,  $c = 3$

$$b^2 - 4ac = 28$$

$b^2 - 4ac$  is positive.

Therefore, this equation has two real solutions:

$$x_1 = (4 - \sqrt{28}) / -2$$

$$x_2 = (4 + \sqrt{28}) / -2$$

That is:

$$x_1 = -2 + \sqrt{7}$$

$$x_2 = -2 - \sqrt{7}$$

or approximately:

$$x_1 = 0.645751311$$

$$x_2 = -4.645751311$$

Notice that in both cases, the system performs courtesy simplifications (using a built-in expert system) in order to give both exact expressions and approximations to the solutions. It also selects coordinates for the graph window that allow optimal viewing of the graph.

A problem object can easily incorporate such a reporting facility, and our next generation of problem objects will be designed to do this. Another example from this Microworld shows what heuristic support may accompany a problem

object associated with quadratic inequalities. It explains the “completion of the square” method.

The student selects the type of inequality and then asks the system to generate a random quadratic inequality by pressing “get  $a$ ,  $b$  and  $c$ ” below. When she selects “graph parabola” the system resizes the screen and draws the parabola. In this case, the inequality to be solved is  $x^2 + 9x > 0$ . When she presses “solve inequality” the step-by-step solution using completion of the square is given in the right-hand window. The complete text is printed below the picture.

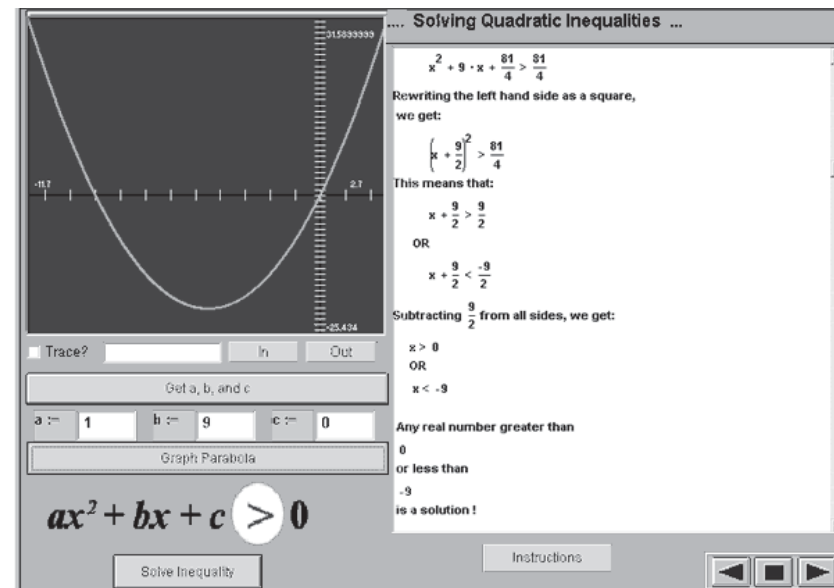


Figure 6:

Solving inequality:

$$x^2 + 9x > 0$$

To complete the square, we add  $81/4$  to both sides:

$$x^2 + 9x + 81/4 > 81/4$$

Rewriting the left hand side as a square, we get:

$$(x + 9/2)^2 > 81/4$$

This means that:

$$x+9/2 > 9/2$$

OR

$$x+9/2 < -9/2$$

Subtracting  $9/2$  from all sides, we get:

$$x > 0$$

OR

$$x < -9$$

Any real number greater than

0

or less than

-9

is a solution!

Now the solution is also depicted graphically in the original window using thick green lines to indicate the intervals as below.

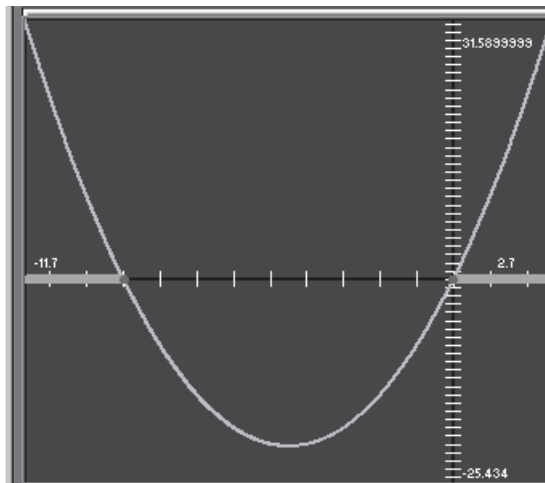


Figure 7:

A final example of a Mathwright Microworld that gives the sort of feedback to the reader that our new problem objects will provide is Dr. Ravinder Kumar's *A Primer on Derivatives*. In this 36-page book, the reader learns many aspects of differentiation, both by following examples, and by doing explorations like the one below on the chain rule.

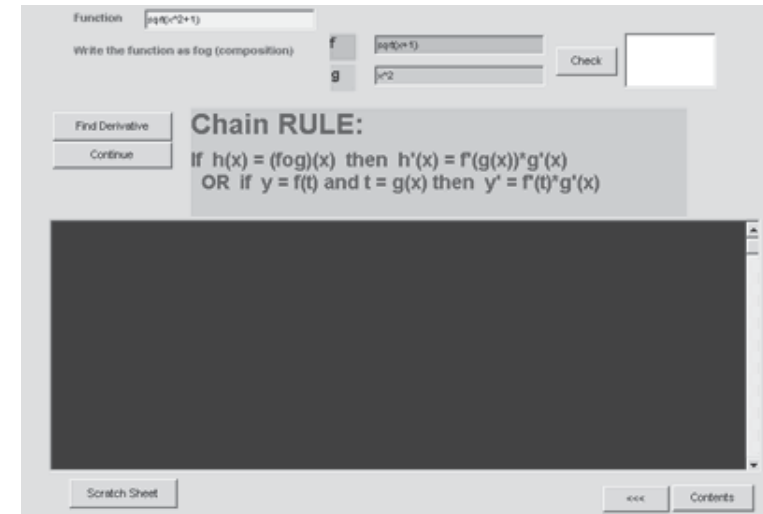


Figure 8:

The first step is for the reader to supply a function (or use the one given) and then to represent it as a composition of two functions.



Figure 9:

When the student checks and sees that her composition was correct, she proceeds to the "Find Derivative" button and sees a step-by-step calculation of the derivative, as below. The full text of that calculation is printed below the picture.

To find the derivative of  $\sqrt{x^2 + 1}$ , we first write it as  $\sqrt{t + 1}$

Then we find the derivative of  $\sqrt{t + 1}$

and the derivative of  $x^2$ , and multiply them and then simplify the product.

Now

$$d(\sqrt{t + 1}) = \frac{1}{2} \cdot \frac{1}{\sqrt{t + 1}}$$

And

$$d(x^2) = 2 \cdot x$$

So

$$d(\sqrt{x^2 + 1}) = \frac{1}{2} \cdot \frac{1}{\sqrt{t + 1}} \cdot 2 \cdot x$$

Therefore

$$d(\sqrt{x^2 + 1}) = \frac{1}{2} \cdot \frac{1}{\sqrt{x^2 + 1}} \cdot 2 \cdot x$$

Simplifying we get

$$d(\sqrt{x^2 + 1}) = \frac{1}{\sqrt{x^2 + 1}} \cdot x$$

Figure 10:

*To find the derivative of  $\sqrt{x^2+1}$ , we first write it as  $\sqrt{t+1}$*

*Then we find the derivative of  $\sqrt{t+1}$*

*and the derivative of  $x^2$ , and multiply them and then simplify the product.*

*Now*

$$d(\sqrt{t+1}) = 1/2 * 1 / \sqrt{t+1}$$

*And*

$$d(x^2) = 2 * x$$

*So*

$$d(\sqrt{x^2+1}) = 1/2 * 1 / \sqrt{t+1} * 2 * x$$

*Therefore*

$$d(\sqrt{x^2+1}) = 1/2 * 1 / \sqrt{x^2+1} * 2 * x$$

*Simplifying we get*

$$d(\sqrt{x^2+1}) = 1 / \sqrt{x^2+1} * x$$

All of the simplifications done in the report are automatic.

While problem objects can be the substrate for pedagogically useful drill and practice environments, it is important to emphasize that the essential feature

of each object is its activation rule. While the framework of the object, created in the problem designer, captures the problem syntax in its template, the activation rule requires a deeper semantic analysis of the problem. Our next step, the addition in each object of a reporting function that provides step-by-step explanation of the solution is clearly more ambitious. In my view, it is in the encoding of problem semantics in the object that both the challenge and the promise of these objects are to be found.

If you have not done so yet, you might like to open the Microworld now and see how these ideas come to life in your browser.

The New Mathwright Library and Cafe contains a growing collection of interactive Microworlds, many of which, in the MATH Cafe, are freely available to Library visitors for them to read in their browsers. Each of these books has also a downloadable offline version that Library members may add to their permanent collections, on any of their computers. If you wish to learn more about Mathwright as an authoring platform, or if you would like to read some of our free Interactive Web Books in your browser before joining, please visit the MATH Cafe.