







































































































































































































































3. How many solutions does the equation  $2x^2 - 3x - 2 = 0$  have?  
 none                      one                      two

### End Quiz

► In the partial verbatim listing that follows, notice the field name has been changed from `qzdiscr11`, which is the name of the quiz previous to his one, to `qzdiscr1f`. Different quizzes must have unique field names.

```
\begin{quiz*}{qzdiscr1f} Using the discriminant,  $b^2-4ac$ ,
respond to each of the following questions.
\begin{questions}
.....
.....
.....
\end{questions}
\end{quiz*}\quad\ScoreField\currQuiz\o1Bdry\CorrButton\currQuiz
```

► Notice that in this example, the `\ScoreField` and the `\CorrButton` are positioned following the ‘End Quiz’; this makes the design more compact and nicer looking.

#### • The `nocorrections` Option

Including the corrections adds quite a bit more JavaScript code to the .pdf document, this feature is ‘on’ by default. If you have a document in which you do not want to have the option of offering corrected quizzes, then just specify `nocorrections` in the option list of `exerquiz`.

There are also a couple of macros you can use to override the option switch, they are `\CorrectionsOn` and `\CorrectionsOff`. Each remains in effect until the other is invoked.

► If the `nocorrections` option is taken, then the `\CorrButton` does not appear for a quiz.

#### • Showing Partial Credit Markup

When a quiz is corrected, it is possible to show how much credit was given.

- 🔗 The demo file for this feature is `pc_test.pdf`.
- 🔗 The example file `pc_test1.pdf` illustrates how to have a multiple selection problem and a grouped problem with no partial credit. The student must get all parts correct to get full credit for that problem, otherwise, no credit is assigned.

Place the command `\showCreditMarkup` before a quiz you want the partial credit markup. The command `\hideCreditMarkup` hides this markup (it switches off the previous `\showCreditMarkup`); place this command before a quiz for which you don’t want the markup. The default is not to show any partial credit markup.

When a question item introduces a series of question, but poses no question itself, place `\multipartquestion` prior to that item, for example



```

\multipartquestion
\item\PTs{6} Answer each of these questions
\begin{question}
  \item\PTs{3} Answer this
  \item\PTs{3} Answer that
\end{question}

```

Other relevant commands are `\aeb@creditmarkup` and `\aeb@creditmarkupfmt`; the first one defines a little text field to hold the partial-credit data and the second one sets the placement of the text field. These can be redefined if the default definitions are not to your liking. The markup in red that appears in the left margin shows the number of points the student received for that problem, for example **3 pts**. The string `pts` (and its singular counterpart, `pt`) is language dependent, in English it refers to “points” (or “point”). You can change this string to an abbreviation of points (`point`) using the commands `\ptsLabel` and `\ptLabel`. The default definitions are `\ptsLabel{pts}` and `\ptLabel{pt}`, respectively.

**Important:** Adobe Reader 5.0 or later is required for this feature.

## 21.6. Quizzes with Solutions

In addition to scoring and marking the quizzes, you can also (optionally) provide solutions as well. To enter a solution to a multiple choice question, use a `solution` environment, and attach a named destination to the `answers` environment. A partial verbatim listing follows the next example.

**Begin Quiz** Answer each of the following. Passing is 100%.

### 1. Who created T<sub>E</sub>X?

Knuth                      Lamport                      Carlisle                      Rahtz

### 2. Who originally wrote E<sub>T</sub><sub>E</sub>X?

Knuth                      Lamport                      Carlisle                      Rahtz

**End Quiz**

After the quiz is completed and the corrections button is pressed, the corrections appear. The correct answer has a green filled circle or a green check; this circle is now outlined by a green rectangle to indicate that this is a link to the solution. Click on the green dot and jump to the solution!

Solutions do not have to appear. Some problems can have solutions, while others do not. The ones with the solutions have the green boundary to indicate a link to the solution.

Here is a partial listing of the above example.

```

\begin{quiz*}{qzTeX} Answer each of the following.
Passing is 100%.
\begin{questions}

```



```
\symbolchoice{diamond}
\everyqRadioButton{\textColor{0 0 1}
  \BC{.690 .769 .871}\BG{.941 1 .941}}
```

**Begin Quiz** Answer each of the following. Passing is 100%.

1. Who created T<sub>E</sub>X?

Knuth                      Lamport                      Carlisle                      Rahtz

2. Who originally wrote L<sup>A</sup>T<sub>E</sub>X?

Knuth                      Lamport                      Carlisle                      Rahtz

**End Quiz**

If desired, we can return to the defaults:

```
\symbolchoice{check} % restore the default
\everyqRadioButton{}
```

► The `\symbolchoice` can also be introduced into the flow of the code through the `\everyqRadioButton`:

```
\everyqRadioButton{\symbolchoice{diamond}\textColor{0 0 1}
  \BC{.690 .769 .871}\BG{.941 1 .941}}
```

After the quiz, we could return to the defaults by

```
\everyqRadioButton{}
```

► See the document [eformman.pdf](#) for complete documentation on how to modify a field using the optional first argument, and how to use the “every” command.<sup>25</sup>

#### • Change color of Correction Marks

The colors used to mark quizzes are changed in the *preamble* by redefining the commands listed below. The default definitions are,

```
\newcommand\righColorJSDef{["RGB", 0, .6, 0]} % color for right response
\newcommand\wrngColorJSDef{color.red} % color for wrong response
\newcommand\partialColorJSDef{color.blue} % color for partial credit
\newcommand\defaultColorJSDef{color.black} % reset color
```

The colors are inserted into the field using JavaScript, so the color definitions are in the color space of the JavaScript color object model.

<sup>25</sup>eforms manual on Web: <http://www.math.uakron.edu/~dpstory/eformman.pdf>



**Begin Quiz** Answer each of the following. Passing is 100%.

1. What TeX System does Thomas Esser maintain?

MiKTeX                      csTeX                      teTeX                      fpTeX

2. What TeX System does Fabrice Popineau maintain?

MiKTeX                      csTeX                      teTeX                      fpTeX

3. What TeX System does Christian Schenk maintain?

MiKTeX                      csTeX                      teTeX                      fpTeX


**End Quiz**

The new part is the customized scoring and correction button. Here is a verbatim listing of the `\ScoreField` and `\CorrButton` macros.

```
\ScoreField[\BC{0 0 1}]{qz:TeXc}\o1Bdry
  \CorrButton[\BC{0 0 1} % blue border color
  \CA{TeX} % Button text
  \RC{Users} % rollover text
  \AC{Group} % pushed text
  \textFont{TiRo} % text font: Times Roman
  \textSize{10} % text size: 10 point
  \textColor{0 0 1} % blue text
  \W{1}\S{I} % border width 1, inset button
  ]{qz:TeXc}
```

-  This example—as well as others—appears in [webeqtst.pdf](#), a demo file from the AcroTeX Blog

## 21.8. `\titleQuiz` and `\fancyQuizHeaders`

-  The demo file for these features is [title\\_quiz.pdf](#).

### • `\titleQuiz`

To assign a title to your quiz use `\titleQuiz`. This title appears just after the ‘Begin Quiz’ button

**Begin Quiz Two Question MC** Answer each of these to the best of your ability.

1. Math is fun.

(a) True                      (b) False

2. Statistics is fun.

(a) True                      (b) False

**End Quiz**

The partial verbatim code for the above quiz follows:









**Begin Quiz** Answer each of the following. Passing is 100%.

- (4<sup>pts</sup>) If  $\lim_{x \rightarrow a} f(x) = f(a)$ , then we say that  $f$  is...  
differentiable          continuous          integrable
- (6<sup>pts</sup>) Name *one* of the two people recognized as a founder of Calculus.

**End Quiz**

Answers:

Points:

Percent:

 See source file for the PDF [quizpts.pdf](#) for a more elaborate version of this question, as well as the source code.

- `\PTs#1`: This macro takes one argument, the number of points to be assigned to the current problem. Place this command immediately after the `\item` in the questions environment. For example, in the above quiz we had

```
\item\PTs{6} Name \emph{one} of the two people recognized
as a founder of Calculus.
```

- `\PTsHook#1`: This macro, which takes one argument, can be used to typeset the points assigned. and is called by `\PTs`. The argument is what is to be typeset. The value assigned the current problem by `\PTs` is contained within the macro `\eqPTs`. In the quiz above, we had

```
\PTsHook{($\eqPTs^{\text{pts}}$)}
```

- There are three other commands that create text fields to display results from a quiz with points assigned:

- `\PointsField[#1]#2`: The number of points earned for the quiz, the total points are also reported. The parameter #2 is the base name of the quiz.
- `\PercentField[#1]#2`: The percentage score for the quiz. The parameter #1 is the base name of the quiz.
- `\GradeField[#1]#2`: The letter grade of the performance on the quiz. The parameter #2 is the base name of the quiz. The values placed in this field are determined by the macro `\eqGradeScale`.

- `\eqGradeScale`: This macro sets the grade scale of a quiz, the default definition is

```
\newcommand\eqGradeScale{"A", [90, 100], "B", [80, 90],
"C", [70, 80], "D", [60, 70], "F", [0, 60]}
```

The ways things are defined now, there can be only one grade scale per document. The value of `\eqGradeScale` is a matrix with an even number of elements. The odd numbered elements are the grades; the even number elements are intervals of percentages (percentages of the total number of points on the quiz). If the percentage of the score falls into a particular range, the corresponding grade is assigned.



Answer these questions truthfully.

1. Do you understanding the concept of `\floatQuiz`?  
 (a) Yes                      (b) No
2. Do you understanding the concept of `\dockQuiz`?  
 (a) Yes                      (b) No

After you have finished the quiz, click on the        button. Click the        button if you wish and your quiz will be marked. Click on the choice with the green border to jump to the solution.

To return to the default behavior of the quiz environment place `\dockQuiz` following the quiz. Quizzes that come after the `\dockQuiz` will then have the standard quiz format.

 See the source file for the PDF [floatquiz.pdf](#) for the  $\LaTeX$  code of the above example.

## 22. Objective Style Questions

Beginning with version 2.0 of `exerquiz`, objective style questions can be posed. Single questions can be posed in the `oQuestion` environment, multiple questions can be placed in either the `shortquiz` or the `quiz` environments. This section discusses this type of question and all of its supporting commands.

### 22.1. Math and Text Questions

`Exerquiz` distinguishes between two types of open ended or objective questions:

1. A mathematical question that requires a mathematical expression as the answer.
  2. A question that requires a text answer.
- The demo file `jquiz.tst.tex` is an important source of examples and instruction for the mathematical type question; additionally, the file `jtxt.tst.tex` has many examples for the text type question.

#### • The Mathematical Question

At this stage in the development of `exerquiz`, a (mathematical) question can be posed that requires an answer that is a function of one or more declared variables  $x$ ,  $y$ ,  $z$ , etc. Thus, when the declared variables  $x$ ,  $y$ ,  $z$  are given a value, the answer is reduced to a number.

For example, the answer to the question “Differentiate  $\frac{d}{dx} \sin^2(x)$ ”, is a function in one variable  $x$ , it can be evaluated numerically and can, therefore, be posed:

































- In this example, another built-in macro, `\sqTallyTotal` was used. This macro creates a text field that accumulates the totals of all the tally boxes.
- ▶ The `shortquiz` environment can also be used for a single objective question. Just don't use the `questions` environment within.

```
\begin{shortquiz}[anExample]
  <an objective style question>
\end{shortquiz}
```

## 22.7. The quiz Environment

Objective questions can be mixed in with multiple choice questions within the `quiz` environment. When posing an objective style question in the `quiz` environment, use `\RespBoxMath` and `\RespBoxTxt`, and optionally include the `\CorrAnsButton` button and the `\AnswerField` text field.

Since the evaluation of the quiz is delayed until the user has finished the quiz, the `\sqTallyBox` macro is not applicable here.

Answer each of the following. Passing is 100%.

1. If  $f$  is differentiable, then  $f$  is continuous.  
     True                      False
2.  $\frac{d}{dx} \sin^2(x) =$
3. Name *one* of the two people recognized as a founder of Calculus.

Answers:

- ▶ The buttons created by `\CorrAnsButton` are hidden until the user ends the quiz (and gets scored) and clicks on the corrections button (`\CorrButton`). The `\CorrAnsButton` should not be included if there is no `\CorrButton`.
- ▶ Pressing on the “Ans” button populates the text field created by `\AnswerField` with the author's correct answer. If there is a solution to the problem, the “Ans” button is outlined in green. Shift-click on the “Ans” button to jump to the solution.
- ▶ The `quiz` environment requires a field name. This same name is used by the objective style question as well.

The listing for the above example follows.

```
\begin{quiz*}{oQq}
Answer each of the following. Passing is 100%.
\begin{questions}
```



but it also makes the corresponding math fill-in read only, so that the student cannot change the answer already provided.

Ideally, the student first enters an answer, and once satisfied with the answer, can then get the correct answer for future questions.

See the demo file [prompt\\_tst.pdf](#) from the [AcroTeX Blog](#) for an example of usage.

#### • Grouped Math/Text Fill-in Questions

Exerquiz defines a grouping environment, `mathGrp`, for math fill-in and *text fill-in* questions where the response to the question might require entering text into multiple math fill-in fields.

When you use the `mathGrp` environment to enclose a set of related math questions, you need to use the `\CorrAnsButtonGrp` button, instead of the `\CorrAnsButton` button. The required argument for this button is a comma delimited list of the answers that appear within the grouped questions. The answers should be listed in the same order  $\TeX$  processes the math (or text) questions. The group of questions is processed as if it were a single question.

► For example...

(3<sup>pts</sup>) Compute the following cross product:

$$(3\vec{i} - 2\vec{j}) \times (\vec{i} + 5\vec{k}) = \quad \vec{i} + \quad \vec{j} + \quad \vec{k}$$

$\underbrace{\hspace{10em}}_{\text{ScoreField}} \quad \underbrace{\hspace{10em}}_{\text{PointsField}}$

Ans:

#### Notes:

- If you miss any one of the three answers, the `ScoreField` reports back ‘Score: 0 out of 1’. There is only one question there, to get it correct, you must answer all three inputs correctly.
- Points can be assigned to the individual responses and a score is given based on the validity of the inputs and the corresponding points. There is a default JavaScript function that scores the results. The author can define a custom JavaScript function to have a more “exotic” method of evaluating the group. See the test file `grp_test.tex` for details.
- Notice that after you take the quiz and click on “Correct” button, the “Ans” button appears (as usual). If you click repeatedly on this “Ans” button, you can cycle through all answers to this question; the response box is highlighted (or put in focus) and the answer appears in the answer field provided.
- See the demo file `grp_test.tex` for the source code of this example, as well as more technical details of the `mathGrp` environment.







### 22.10. The correct-type math problem


The traditional way of posing a mathematical objective-style question is to use the `\RespBoxMath` command, which presents an empty text field for the student to enter the answer. However, an alternate approach is to pre-populate the field and ask the student to edit the expression provided to obtain a correct answer.

Edit the expression to obtain the correct answer:

$$\frac{d}{dx} \sin(2x^2) =$$

```
\RespBoxMath[\DV{\cos(4x)}\V{\cos(4x)}]
{4xcos(2x^2)}{2}{.001}{[0,1]}
```

The field is initially populated by an incorrect answer (a very common mistake), the student needs to correct the answer.

-  Refer to the AcroTeX Blog article [Correcting a math problem](#) for additional examples. The article also discusses the use of multi-letter variables in this context of a correct-type question.

### 22.11. Evaluating Equivalent Expressions

The `\RespBoxMath` macro, and its underlying JavaScript code, is designed to determine the correctness of an answer, numerical or symbolic. We can pose such questions as “find the derivative” of this function or “find the (definite or indefinite) integral” of that function, and so on. This is all accomplished by randomly choosing numbers from a specified interval and evaluating both the user’s and author’s answer to the question; if they are close to each other, then the user’s answer is called correct. The system works pretty well.

The `exerquiz` scheme does not perform well with such questions as “fully simplify the radical  $\sqrt{72}$ ,” or “factor the polynomial  $2x^2 - 4x + 2$ ,” in both cases, the student can simply type in the given expression and `exerquiz` will call them equal. Indeed it is true that  $\sqrt{72}$  equates to  $\sqrt{72}$  and the polynomial  $2x^2 - 4x + 2$  equates to  $2x^2 - 4x + 2$ , as is shown by evaluating randomly selected numbers from an interval, `exerquiz`-style!

To illustrate, we use the problem of simplifying  $\sqrt{72}$ .

**Quiz** Simplify the following radical expression by factoring out all perfect squares.

1. Simplify  $\sqrt{72}$  by factoring out all perfect squares.

$$\sqrt{72} =$$

`Exerquiz` performs its magic by a combination of the usual `exerquiz` processing and new pre- and post-parsing techniques, where appropriate code is passed through of the ninth parameter of `\RespBoxMath`.

The verbatim listing follows.







The `\preReqForm` only searches for the specified patterns. For example, the responses `\sin(\pi/2)6\sqrt{2}` and `6\sqrt{2}x` pass both the `\preDenyForm` (no digits follow the square root), and pass the `\preReqForm` because both answers have (or do not have) the patterns; in the former case, the answer is judged correct, and in the latter case, incorrect. You can try to improve the pattern like so

```
\/\rebstr\rechrclass{+-}*\redigit+sqrt\refac{\reany+}\reestr/
```

The `\rebstr` (begin string anchor) expands to ‘`^`’ and represents the beginning of the line, while `\reestr` (end string anchor) expands to ‘`$`’ represents the end of the line. In this case, the user’s answer is required to begin with an optional plus or minus, followed by one or more digits, followed by a square root, followed by the end of the line.

Here is that same short quiz with the above changes (and `*` replaced by `?`).

**Quiz** Simplify the following radical expression by factoring out all perfect squares.

1.  $\sqrt{72}$

$$\sqrt{72} =$$

Now try `\sin(\pi/2)6\sqrt{2}` or `6\sqrt{2}x`, and other aberrant variations.

**The postParse property.** After the `priorParse` stage, `exerquiz` checks the answer for correctness, against the author’s answer, using randomly selected numbers. After that comes the `postParse` event. The syntax is the same except there is no warning message.

```
\preReqForm(regex|array_regexp)
```

If the user’s *answer is correct* at this stage *and* fits into one of the patterns as defined by the regular expressions, the answer will then be marked wrong.

Let’s look at the post parse code for this example, from line (10) on page 136, we see,

```
10 postParse:\Array(\postDenyForm(/sqrt(\refac{72}|\refac{8})))%
```

Even though there is not an array of filter functions, a filter function with arguments must always be enclosed in an array.

Within the array, we execute our `\postDenyForm`, it has only one argument, a single regular expression: `/sqrt(\refac{72}|\refac{8})/`.<sup>29</sup> There are three “correct answers” to this problem, they are `\sqrt{72}`, `3\sqrt{8}`, and `6\sqrt{2}`, this latter one being the only correct answer to this question, for the instructions were to factor out *all* perfect squares. The regular expression here catches any answer containing `\sqrt{72}` or `3\sqrt{8}` and marks it wrong. :- (

<sup>29</sup>The use of alternation here is not needed, an answer involving `\sqrt{72}` would have been filtered out by the `\preReqForm` filter, do you know why?

























































-  Refer to the AcroTeX Blog article *The dljslib package: German number format* for more information and working examples.

### 29.3. Compare Functions

Compare functions are used to compare the author's answer with the student's answer. Based on the comparison, the student's answer is judged correct, or not. The `exerquiz` package has two built-in compare functions: `diffCompare` (the default) and `relDiffCompare`. The former makes the comparison based on absolute differences

$$\text{diffCompare} = |\text{Author Ans} - \text{Student Ans}|$$

while the latter uses absolute relative differences to make the comparison

$$\text{relDiffCompare} = \frac{|\text{Author Ans} - \text{Student Ans}|}{|\text{Author Ans}|}$$

Comparison functions are passed through the 9<sup>th</sup> argument of `\RespBoxMath`, if no such argument is passed, `diffCompare` is used.

- **indefIntegral**


The answer to an indefinite integral is non-unique; however all answers differ by a constant. This compare function is used with indefinite integrals. Note the name of the function is `indefCompare`, this is the name you use to call the function. To use this compare function, insert its name in optional 9<sup>th</sup> argument,

```
\RespBoxMath[#1]#2(#3)[#4]#5#6#7#8[#9]*#10
```

that's the argument that follows the interval specification.

An example of usage of this function is found in the sample file `dljslib_ex.tex`, referenced below; from that file we have:

```
\displaystyle\int\sin(x)\,dx =
\RespBoxMath{-cos(x)}{4}{.0001}{[0,1]}[indefCompare]$
```

-  See the demo file `dljslib_ex.pdf` for an example of indefinite integration comparison.

- **satisfyEq**

The `satisfyEq` library of procedures supports the type of question where the student is asked to enter one or more points that satisfy an equation.

We pose a problem where the student is asked to enter an order pair or triple that satisfies an equation. For example

- ▶ Enter a point that lies on the line  $2x + 3y = 6$ , the answer is an ordered pair of numbers, for example (1, 2).

A point is

The verbatim listing follows:

```

1 \RespBoxMath[\rectW{1.5in}\textSize{0}]{2x+3y-6}(xy){1}
2   {.0001}\viidna}*{ProcRespEvalEq}\kern1bp
3   \CorrAnsButton{various, such as (0,2)}

```

Line (2) shows the placement of one of the procedures, `ProcRespEvalEq`, that is part of `satisfyEq`.

There are four procedures in this set:


1. `ProcRespEvalEq`: Supports a single equation, the answer is a single ordered  $n$ -tuple, the entries are allowed to be zero.
2. `ProcRespEvalEqNonZero`: Supports a single equation, the answer is a single ordered  $n$ -tuple, the entries are *not allowed* to be zero.
3. `ProcRespEvalEqList`: Supports a single equation, the answer consists of  $k$ ,  $n$ -tuples, the components of the  $n$ -tuples are allowed to be zero. Each  $n$ -tuple is separated by a semi-colon (;).
4. `ProcRespEvalEqListNonZero`: Supports a single equation, the answer consists of  $k$ , ordered  $n$ -tuples, the components of the  $n$ -tuples are *not allowed* to be zero. Each  $n$ -tuple is separated by a semi-colon (;).

**Some Technical Details.** Recall the parameter list for `\RespBoxMath`,

```
\RespBoxMath[#1]#2(#3)[#4]#5#6#7#8[#9]*#10
```

`ProcRespEvalEq`, *et al*, are not based on random point generation, as are all the other `exerquiz` functions. Here, the user enters numerical data, the function then verifies the data entered satisfies the given equation. The role of one of the parameters has changed, others are ignored:

- The role of parameter `#2` (the correct answer) has changed. Suppose the equation presented to the user is  $F = G$ .
  - For `ProcRespEvalEq` and `ProcRespEvalEqNonZero`, pass  $F - G$  as parameter `#2`.
  - For `ProcRespEvalEqList` and `ProcRespEvalEqListNonZero`, pass  $k;F-G$  as the parameter `#2`. For this type of problem, the author is asking the user to enter  $k$ ,  $n$ -tuples that satisfy the equation. Note  $k$  and  $F-G$  are delimited by a semicolon (;).
- The value of `epsilon` (parameter `#6`) is obeyed.
- Parameter `#5`, the number of iterations, is not used. Though this parameter is not used, `#5` is required by the macro, so just use 1 as the number of iterations.
- Parameter `#7`, the domain of the variables, is not used. Again, since this is a required parameter, use `\viidna`; `\viidna` is a special command that can be used for this parameter, in this context only.
- The optional parameter `#9` is not used and is ignored if present.

 Additional examples will (eventually) appear on the [AcroTeX Blog](#).

### 29.4. Filter User's Responses

Filters are JavaScript functions that process the user input before being passed to the response function. Filters are passed to `\RespBoxMath` through the 9<sup>th</sup> argument,

$$\int \sin(x) dx = \text{\RespBoxMath}\{-\cos(x)\}\{4\}\{.0001\}\{[0,1]\}\{\text{indefCompare}\}\$$$

This 9<sup>th</sup> argument can take on two formats: (1) the name of the compare function, as illustrated in '`indefIntegral`' above; (2) a JavaScript object. Use a JavaScript object as the 9<sup>th</sup> argument to specify a filter.

The JavaScript object that can be passed in the 9<sup>th</sup> argument has two properties `comp` and `priorParse`. The object has the form

```
{ comp: <compare>, priorParse: <JS_func> | <array_JS_funcs> }
```

The first property is used to specify a compare function, the second property specifies a filter or filters. The value of `priorParse` may be a JavaScript function to filter the user's response, or an array of JavaScript functions to filter the user input. (The array must be specified using the `\Array` command, illustrated below, see the paragraph '`NoProducts`', on page 168.)

- **nodec**

The `nodec` option defines the JavaScript function `nodec` that attempts to prohibit the use of decimal numbers in the response. The function simply searches the user response for any occurrence of the decimal point. (The German language uses the comma as a decimal point, you can set the decimal point to some other symbol that `nodec` searches for through the command `\setdecimalpoint`, the default definition is `\setdecimalpoint{.}`.)

$$\$ 0.75 = \text{\RespBoxMath}\{3/4\}\{1\}\{.0001\}\{[0,1]\}\{\text{priorParse: nodec}\}\$$$

Alert box message: May be redefined.

```
\newcommand\nodecAlertMsg{%
  "A decimal answer is not acceptable here
  Please express your answer using fractions, square roots,
  e, log, etc."}
```


- **noBinFac**

The `noBinFac` option defines the JavaScript filter function `noBinFac`, that disallows binomial coefficients and factorials in math fill-in questions. Binomial and factorial functions are defined through the `combinatorics` option, page 170.

The number of ways to choose four object from a set of ten is

$$\text{\RespBoxMath}\{C(10,4)\}\{1\}\{.0001\}\{[0,1]\}\{\text{priorParse: noBinFac}\}\text{\CorrAnsButton}\{C(10,4)\}\*\{\text{EvalCorrAnsButton}\}$$

The `noBinFac` filter is used, so the student is not allowed to enter binomial coefficients; however, the document author can use binomial coefficients, note the use in the answer (first argument) and in the `\CorrAnsButton`. The special syntax used for `\CorrAnsButton`, [‘Including a Solution’](#) on page 124.

 Illustrations of this filter can be found in [dljs\\_comb.pdf](#).

Alert box messages: May be redefined.

```
\newcommand\noBinFactBinCoeffAlertMsg{%
  "You may not use this notation here.
  Please evaluate the binomial coefficient.
  You may present your answer as a product rather
  than calculating a very large number."}
\newcommand\noBinFactPermAlertMsg{%
  "You may not use this notation here.
  Please evaluate the permutation.
  You may present your answer as a product rather
  than calculating a very large number."}
\newcommand\noBinFactFactAlertMsg{%
  "You may not use this notation here.
  Please evaluate the factorial.
  You may present your answer as a product rather
  than calculating a very large number."}
```

#### • `limitArith`

The `limitArith` option defines a number of filter functions:

```
DecimalsOnly, NoProducts, NoDivision, NoAddOrSub,
NoArithAllowed, NoExpAllowed, NoTrigAllowed, NoTrigLogAllowed
```

 Examples of these filtering functions can be found in the demo file [limarth.pdf](#).

**DecimalsOnly** The `DecimalsOnly` is a function that takes only numbers, either integer or decimal. Usage

```
$ 2.3 + 4.5 = \RespBoxMath{6.8}{1}{.0001}{{0,1}}
  [{priorParse: DecimalsOnly } ]$
```

**NoProducts** The function `NoProducts` requires that `ImplMulti` is taken. An example of usage is

```
$ 3/4 = \RespBoxMath{.75}{1}{.0001}{{0,1}}
  [{priorParse: \Array(NoDivision, NoProducts) } ]$
```

In the above example, we first call `NoDivision` then `NoProducts`. Several filtering functions can be executed by putting them into an array, using the `\Array` command.<sup>35</sup>

<sup>35</sup>The `\Array` command was originally named `\array`, it has been renamed to avoid clashing with the `amsmath` package.



**NoAddOrSub** This function, as the name implies, denies the use of both addition and subtraction. The algorithm I use here is that if a plus or minus follows anything but a left parenthesis, we assume it is a binary operation of addition or subtraction, respectively.

```
$ 5.1 - 3.2 =
  \RespBoxMath{1.9}{1}{.0001}{[0,1]}[{\priorParse: NoAddOrSub}]$
```

**NoArithAllowed** Deny user the use of all arithmetic functions, including exponents.

```
$ 6/5 = \RespBoxMath{1.25}{1}{.0001}{[0,1]}
  [priorParse: NoArithAllowed]$
```

**NoExpAllowed** Deny user the use of exponents.

```
$ 25^2 = \RespBoxMath{625}{1}{.0001}{[0,1]}
  [priorParse: NoExpAllowed]
```

**NoTrigAllowed** Deny user the use all trig functions, as well as the constants PI and pi, which are aliases for  $\pi$ . Example:

```
$ \sin(\pi/4) = \RespBoxMath{sqrt(2)/2}{1}{.0001}{[0,1]}
  [priorParse: NoTrigAllowed ]$
```

**NoTrigLogAllowed** Deny the use of trig and log functions.

```
$ \ln(e^2)\sin(\pi/4) = \RespBoxMath{sqrt(2)}{1}{.0001}{[0,1]}
  [priorParse: NoTrigLogAllowed ]\CorrAnsButton{sqrt(2)}$
```

**sciNotResp** Allow only a response in *normalized* scientific notation:  $aE([+|-]d)$ , where  $1 \leq |a| \leq 9$ , and  $d$  is one or more digits (0-9) prefixed by a -, or an optional +.

```
Convert 123.4 to scientific notation.
\begin{equation*}
  123.4 = \RespBoxMath{1.234E+2}{*}{1}{0}{[0,1]}[%
    {priorParse: sciNotResp }]\CorrAnsButton{1.234E+2}
\end{equation*}
```

**noNegExp** Does not allow negative exponents. For example,

```
\item Simplify $\displaystyle\frac{4}{r^4}\cdot\frac{r^3}{12}=
  \RespBoxMath{1/(3r)}{r}{*}{3}{.0001}{[0,1]}[{\%
    priorParse:\Array( noDec, noNegExp )
  }]\CorrAnsButton{1/(3r)}$
```

**Alert box messages:** May be redefined.

```
\newcommand\DecimalsOnlyErrorMsg{"Enter only an integer,
  e.g., 17, or a decimal number, e.g. 12.4.
  Using arithmetic operations or built in function is
  not acceptable for this problem."}
\newcommand\NoProductsErrorMsg{"Multiplication is not allowed
```

```

    for this problem."}
\newcommand\NoDivisionErrorMsg{"Division is not allowed
for this problem."}
\newcommand\NoAddOrSubErrorMsg{"Neither addition nor subtraction
is not allowed for this problem."}
\newcommand\NoExpAllowedErrorMsg{"The use of exponents is not
allowed for this problem."}
\newcommand\NoTrigAllowedErrorMsg{"The use of trig functions
in this problem is not allowed."}
\newcommand\NoTrigLogAllowedErrorMsg{"The use of trig and log
functions is not allowed in this problem."}
\newcommand{\sciNotSyntaxError}{"Enter the answer in scientific
notation."}
\newcommand{\sciNotNormalForm}{"The scientific notation entered
is not in normalized form."}
\newcommand{\NoNegExpMsg}{"No negative exponents permitted,
keep working!"}

```

- **useEnNums**

This is an analog to `useDeNums`, but no new response function is defined. It deals with filtering the users' responses to a specified number of decimal places.

 Refer to the AcroTeX Blog article *The dljslib package: English number format* for more information and working examples.

## 29.5. Extending Numerical Functions

These JavaScript functions extend the collection of built-in functions that perform numerical calculations.


- **combinatorics**

This option defines three JavaScript functions: a binomial coefficient function `ch(n, r)`; a number of permutations function `perm(n, r)`, and a factorial function `fact(n)`.

**User Accessible Functions** The user should use the names `C`, `P` and `fact` to access the functions `ch`, `perm` and `fact`, respectively:

- $C(n, r) = \text{ch}(n, r) = \binom{n}{r} = \frac{n!}{r!(n-r)!}$
- $P(n, r) = \text{perm}(n, r) = n \cdot (n-1) \cdots (n-r)$
- $\text{fact}(n) = n \cdot (n-1) \cdots 3 \cdot 2 \cdot 1$

**Note:** If you place `\replaceExclPt{true}` in the preamble, the user is allowed to enter factorials using the traditional exclamation notation: for example, `17!` is the same as `fact(17)`, and `(17)!`, `[17]!` or `{17}!` are all interpreted as `fact(17)`.

 Illustrations of the `combinatorics` option can be found in [dljs\\_comb.pdf](#).

### 30. List of Options

Options of the Web/Exerquiz Packages	
Options of the Web Package	
<code>dvipsone</code>	dvi-to-ps driver by Y&Y, Inc.
<code>dvips</code>	dvi-to-ps driver
<code>pdftex</code>	tex-to-pdf application
<code>dvipdfm</code>	dvi-to-pdf application
<code>textures</code>	the Textures System for Mac
<code>designi</code> , <code>designii</code> , <code>designiii</code> , <code>designiv</code> , <code>designv</code>	these set screen design parameters
<code>usetemplates</code>	this option activates mechanism for creating colored backgrounds and overlays
<code>leftpanel</code>	creates a left navigation panel
<code>rightpanel</code>	creates a right navigation panel
<code>navibar</code>	inserts a menu bar at the bottom of each page
<code>latextoc</code>	displays the standard toc
<code>usedirectory</code>	causes the title page directory to appear on the title page
<code>forpaper</code>	this turns off color, and does not put solutions on separate pages.
<code>forcolorpaper</code>	Same as <code>forpaper</code> , but does not turn off color, useful for color printers.
<code>latexlayout</code>	web uses page layout for <code>article</code> class. For use with <code>forpaper</code> .
<code>tight</code>	Redefines list environment parameters so lists don't take up so much space
<code>pro</code>	Causes web to input enhanced control over various web elements
<code>nobullets</code>	Forces the use of numbers for subsections, also executes the <code>latextoc</code> option
<code>usesf</code>	switches to sans serif as the default font, good for presentations
<code>unicode</code>	Passes the <code>unicode</code> option to <code>hyperref</code> .
<code>useui</code>	Passes the <code>useui</code> option to <code>eforms</code> .

Options of the Web/Exerquiz Packages (cont.)	
<code>xhyperref</code>	Causes web not to include the hyperref package.
<code>dutch</code>	Dutch for web, passed to exerquiz. (Thanks to Henny Wilbrink)
<code>french</code>	French for web, passed to exerquiz. (Thanks to Jean-Michel Sarlat)
<code>german</code>	German for web, passed to exerquiz. (Thanks to Michael Wiedmann)
<code>italian</code>	Italian for web, passed to exerquiz. (Thanks to PierLuigi Zezza)
<code>norsk</code>	Norwegian for web, passed to exerquiz. (Thanks to Hans Fredrik Nordhaug)
<code>russian</code>	Russian for web, passed to exerquiz. (Thanks to Sergei V. Znamenskii)
<code>spanish</code>	Spanish for web, passed to exerquiz. (Thanks to Pedro Luis Luque)
<code>polish</code>	Polish for web, passed to exerquiz. (Thanks to Jerzy Mycielski)
<code>finnish</code>	Finnish for web, passed to exerquiz. (Thanks to Päivi Porras)
<code>catalan</code>	Catalan for web, passed to exerquiz. (Thanks to Ramon Ballester)
<code>czech</code>	Czech for web, passed to exerquiz. (Thanks to Robert Marik)
<code>brazil</code>	Brazilian Portuguese for web, passed to exerquiz. (Thanks to Koichi Sameshima)
<code>turkish</code>	Turkish for web, passed to exerquiz. (Thanks to Mahmut Koçak)
Options of the Exerquiz Package	
<code>nosolutions</code>	removes the solutions to the exercises
<code>noquizsolutions</code>	removes the solutions to the quizzes
<code>nohiddensolutions</code>	overrides the 'h' (hidden) option for the exercises.
<code>noHiddensolutions</code>	overrides the 'h' and 'H' (hidden) options for the exercises.
<code>nocorrections</code>	removes the ability to correct the quizzes
<code>solutionsafter</code>	solutions to exercises are typeset just after the question

Options of the Web/Exerquiz Packages (cont.)	
<code>forpaper</code>	same function as in <code>web</code> . Needed when <code>exerquiz</code> is not used with <code>web</code>
<code>forcolorpaper</code>	same function as in <code>web</code> . Needed when <code>exerquiz</code> is not used with <code>web</code>
<code>preview</code>	shows the outline of all form fields in the dvi previewer
<code>nodljs</code>	turns off the insertion of DLJS
<code>exercisonly</code>	if document has only exercises, no doc level JS needed
<code>proofing</code>	mark the correct answers for <code>shortquiz</code> & <code>quiz</code> for proof reading.
<code>showgrayletters</code>	Show gray letters under check boxes
<code>allowrandomize</code>	Allow the randomization of choices in multiple choice questions
<code>unicode</code>	Passes the <code>unicode</code> option to <code>hyperref</code> .
<code>useui</code>	Passes the <code>useui</code> option to <code>eforms</code> .
<code>usesumrytbls</code>	Inputs the code used to create quiz summary tables, <a href="#">Section 24</a> , page 143.
<code>contsolns</code>	Places a 'continued' message in the right footer on a solution page.
<code>dutch</code>	JavaScript messages in Dutch (Thanks to Henny Wilbrink)
<code>french</code>	JavaScript messages in French (Thanks to Jean-Michel Sarlat)
<code>german</code>	JavaScript messages in German (Thanks to Michael Wiedmann)
<code>italian</code>	JavaScript messages in Italian (Thanks to PierLuigi Zezza)
<code>norsk</code>	JavaScript messages in Norwegian (Thanks to Hans Fredrik Nordhaug)
<code>russian</code>	JavaScript messages in Russian (Thanks to Sergei V. Znamenskii)
<code>spanish</code>	JavaScript messages in Spanish (Thanks to Pedro Luis Luque)
<code>polish</code>	JavaScript messages in Polish (Thanks to Jerzy Mycielski)
<code>finnish</code>	JavaScript messages in Finnish. (Thanks to Päivi Porras)

Options of the Web/Exerquiz Packages (cont.)	
<code>catalan</code>	JavaScript messages in Catalan. (Thanks to Ramon Ballester)
<code>czech</code>	JavaScript messages in Czech. (Thanks to Robert Marik)
<code>brazil</code>	JavaScript messages in Brazilian Portuguese. (Thanks to Koichi Sameshima)
<code>turkish</code>	JavaScript messages in Turkish. (Thanks to Mahmut Koçak)

## Solutions to Exercises

**Exercise 1.** We evaluate by *integration by parts*:

$$\begin{aligned}
 \int x^2 e^{2x} dx &= \frac{1}{2} x^2 e^{2x} - \int x e^{2x} dx && u = x^2, dv = e^{2x} dx \\
 &= \frac{1}{2} x^2 e^{2x} - \left[ \frac{1}{2} x e^{2x} - \int \frac{1}{2} e^{2x} dx \right] && \text{integration by parts} \\
 &= \frac{1}{2} x^2 e^{2x} - \frac{1}{2} x e^{2x} + \frac{1}{2} \int e^{2x} dx && u = x^2, dv = e^{2x} dx \\
 &= \frac{1}{2} x^2 e^{2x} - \frac{1}{2} x e^{2x} + \frac{1}{4} e^{2x} && \text{integration by parts} \\
 &= \frac{1}{4} (2x^2 - 2x + 1) e^{2x} && \text{simplify!}
 \end{aligned}$$

□

**Exercise 2.**

$$x + y = 1$$

□

**Exercise 3(a)** Velocity is the rate of change of position with respect to time. In symbols:

$$v = \frac{ds}{dt}$$

For our problem, we have

$$v = \frac{ds}{dt} = \frac{d}{dt}(t^2 - 5t + 1) = 2t - 5.$$

The velocity at time  $t$  is given by  $v = 2t - 5$ .

□

**Exercise 3(b)** Acceleration is the rate of change of velocity with respect to time. Thus,

$$a = \frac{dv}{dt}$$

For our problem, we have

$$a = \frac{dv}{dt} = \frac{d}{dt}(2t - 5) = 2.$$

The acceleration at time  $t$  is constant:  $a = 2$ .

□

**Exercise 4(a)**  $i^2 = -1$

□

**Exercise 4(b)**  $i^3 = ii^2 = -i$

□

**Exercise 4(c)**  $z + \bar{z} = \operatorname{Re} z$

□

**Exercise 4(d)**  $\frac{1}{z} = \frac{1 \bar{z}}{z \bar{z}} = \frac{\bar{z}}{z \bar{z}} = \frac{\bar{z}}{|z|^2}$

□

**Exercise 6(a)**  $v = 2t - 5$ .

□

**Problem 8.** This is the solution.

◀

**Exercise 9.** It is well known that  $2 + 2 = 4$ .

□

**Project Hint:** There, you didn't need my help after all.

◀



## Solutions to Quizzes

**Solution to Quiz:** The answer is 'Yes'. The definition requires that

$$F'(x) = f(x) \quad \text{for all } x,$$

well, let's check it out.

The definition of  $f$  is  $f(s) = 4s^3$  and so  $f(x) = 4x^3$ . The definition of  $F$  is  $F(t) = t^4$  and so, by the rules of differentiation,  $F'(t) = 4t^3$ . Thus,  $F'(x) = 4x^3$ . Therefore,

$$F'(x) = 4x^3 = f(x) \quad \text{for all } x,$$

as required by the definition. ■

**Solution to Quiz:** If you erred on this one, more than likely it was on the appropriate multiplicative constant: 6 not 18. At least that's what I'm betting on.

The instructions of the LCD Algorithm said to *completely factor the denominator*. Here's a list of the factors

$$\underbrace{3, x^{3/2}, y^2}_{\text{first term}}, \underbrace{2, 3, x, y^4}_{\text{second term}}$$

Let's rearrange them

$$2, 3, 3, x, x^{3/2}, y^2, y^4$$

Now drop duplicate factors—that's the 3. Oops! I did mention dropping identical factors, didn't I?

$$2, 3, x, x^{3/2}, y^2, y^4$$

Now, group together all terms which have the same base, then drop, from each of these groups all terms but the one with the highest power. We obtain then,

$$2, 3, x^{3/2}, y^4$$

The LCD is the product of same:

$$\text{LCD} = (2)(3)x^{3/2}y^4 = 6x^{3/2}y^4.$$

*Solution Notes:* Alternative (a) will work as a common denominator, but it is not the least common denominator. If you use (a), you will be working with larger numbers than is really necessary. ■

**Solution to Quiz:** Columbus sailed the ocean in 1492. It rhymes. ■

**Solution to Quiz:** The answer is true, of course, math is fun! ■

**Solution to Quiz:** The answer is false, statistics is interesting but not fun. ■

**Solution to Quiz:** Yes, Donald Knuth was the creator of  $\text{\TeX}$ . ■

**Solution to Quiz:** Yes, Leslie Lamport was the creator of  $\LaTeX$ . ■

**Solution to Quiz:** The answer is true, of course, math is fun! ■

**Solution to Quiz:** The answer is false, statistics is interesting but not fun. ■

**Solution to Quiz:** You can't understand it until I explain it! ■

**Solution to Quiz:**

$$\frac{d}{dx} \sin^2(x) = 2 \sin(x) \cos(x) = \sin(2x)$$

**Solution to Quiz:**

$$\frac{d}{dx} \sin^2(x) = 2 \sin(x) \cos(x) = \sin(2x)$$

**Solution to Quiz:** Yes, Isaac Newton and Gottfried Leibniz are considered founders of Calculus. ■

**Solution to Quiz:**

$$\frac{d}{dx} \sin^2(x) = 2 \sin(x) \cos(x) = \sin(2x)$$

**Solution to Quiz:** Yes, Isaac Newton and Gottfried Leibniz are considered founders of Calculus. ■

## References

We list cited referenced documents, articles, and webpages, as well as recommended reading. Cited material has the page number where that entry was referenced.

- [1] Acrobat JavaScript Scripting Reference, Technical Note #5431, Version 6.0., Adobe Systems, Inc. See page 149. <http://partners.adobe.com/asn/acrobat/docs.jsp>
- [2] Leslie Lamport, *LaTeX: A Document Preparation System* (2nd ed.), Addison-Wesley Publishing Company, 1994, ISBN 0-201-52983-1.
- [3] Michel Goossens, Frank Mittelbach and Alexander Samarin, *The LaTeX Companion*, second ed., Addison-Wesley Publishing Company, 2004, ISBN 0-201-36299-6. See page 74.
- [4] Michel Goossens, Sebastian Rahtz, and Frank Mittelbach, *The LaTeX Graphics Companion*, Addison-Wesley Publishing Company, 1997, ISBN 0-201-85469-4.
- [5] Michel Goossens, and Rahtz, Sebastian, with Gurari, Eitan, Moore, Ross, and Sutor, Robert, *The LaTeX Web Companion*, Addison Wesley Longman, Reading, Massachusetts, USA, 1999. ISBN: 0-201-43311-7. See page 24.
- [6] Helmut Kopka and Patrick W. Daly, *A Guide to LaTeX2e* (2nd ed.), Addison-Wesley Publishing Company, 1995, ISBN 0-201-43777-X.
- [7] Donald E. Knuth, *The TeXbook*, Addison-Wesley Publishing Company, 1987, ISBN 0-201-13448-9.
- [8] Thomas Merz, *Web Publishing with Acrobat/PDF*, Springer-Verlag Berlin Heidelberg New York, 1998, ISBN 3-540-63762-1.
- [9] D.P. Story, *Pdfmarks: Links and Forms*, AcroTeX Web Site, 1998.  
[www.math.uakron.edu/~dpstory/acrotex.html](http://www.math.uakron.edu/~dpstory/acrotex.html)
- [10] D.P. Story, *Using LaTeX to Create Quality PDF Documents for the WWW*, AcroTeX Web Site, 1998. See page 18. [www.math.uakron.edu/~dpstory/acrotex.html](http://www.math.uakron.edu/~dpstory/acrotex.html)
- [11] D. P. Story, *AcroTeX eDucation System Tools: LaTeX for interactive PDF documents*, in preparation. See pages 9 and 155.