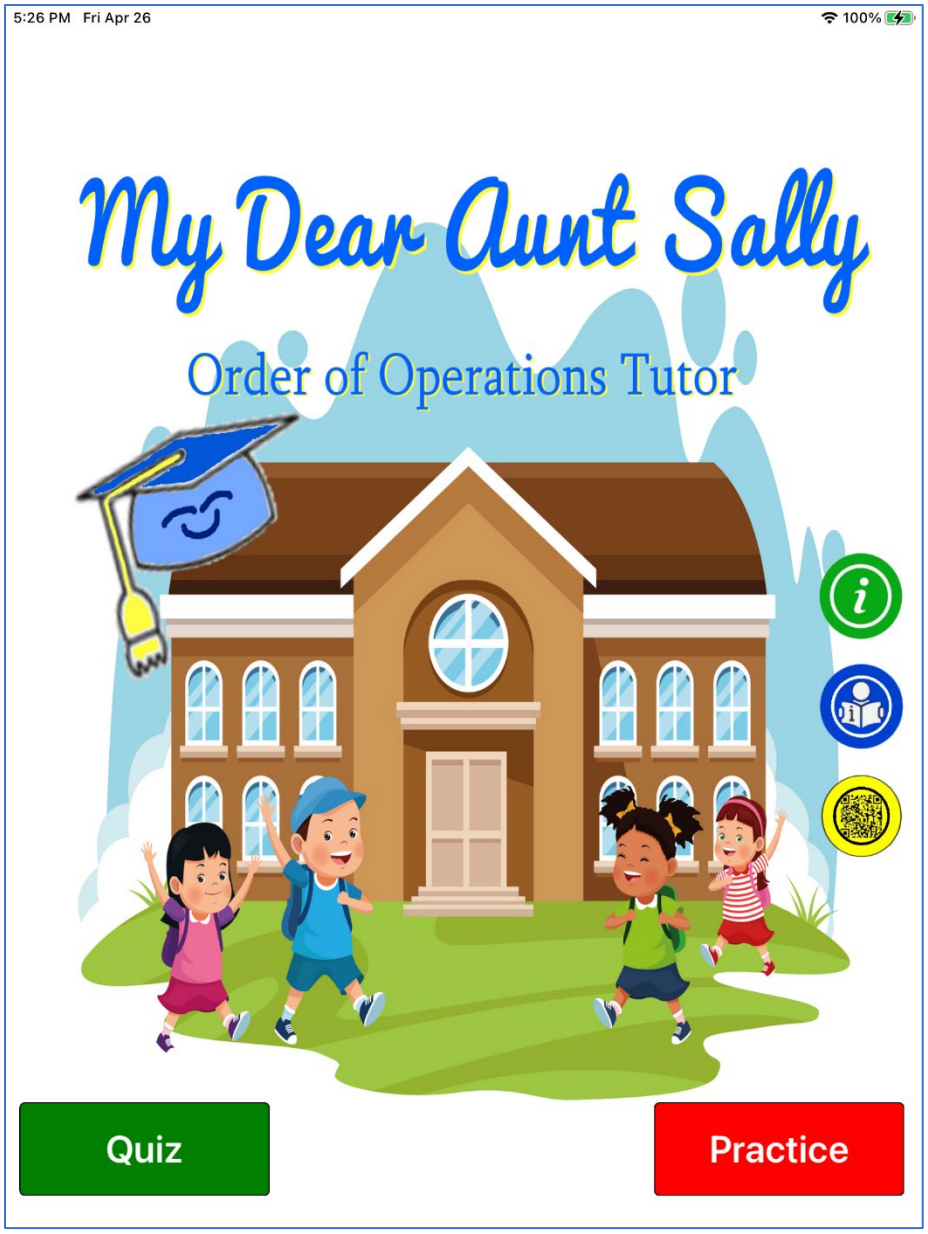
My Dear Aunt Sally

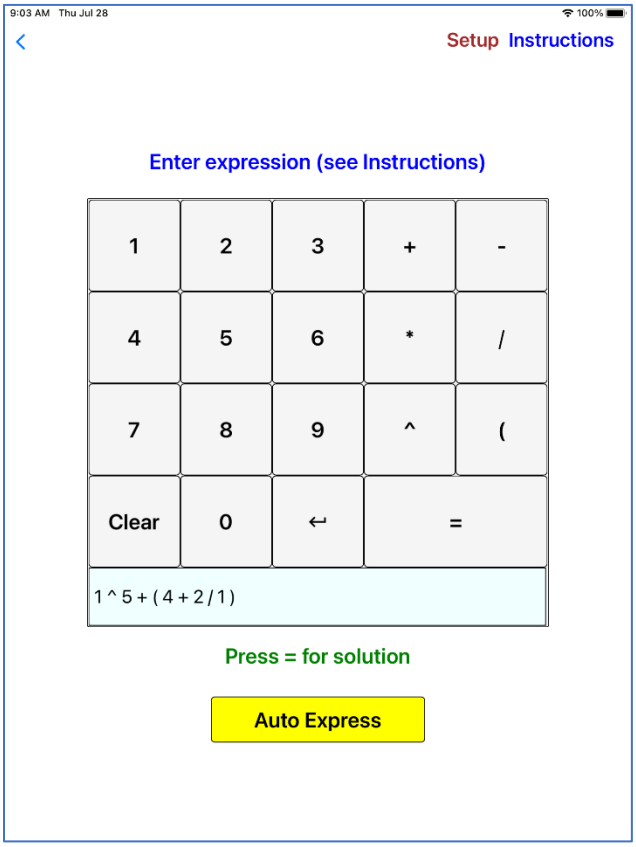
Order of Operations Tutor

USER'S GUIDE

The iPad opening screen.



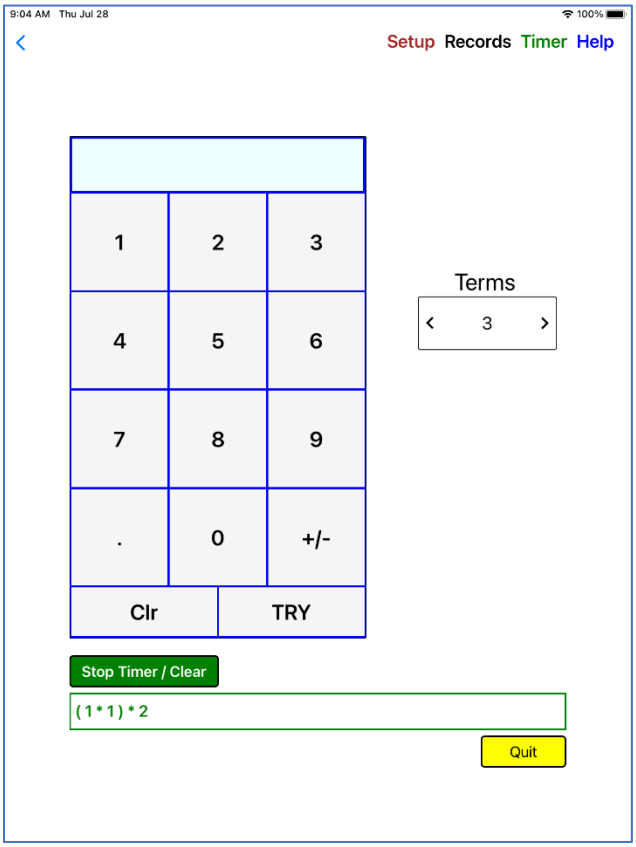
I. OPERATIONAL MODES



In the **PRACTICE** mode, algebraic expressions are entered free hand using an internal keyboard which limits entries to the five common mathematical operators (+ - / * ^) as well as one level of parenthesis. There is a maximum term limit of 10 variables, but no limit to their magnitude in this mode.

Practice sessions are untimed and ungraded.

These sessions do not require pencil and paper. An external keyboard is not needed.



In the **QUIZ** mode:

(a) 2 - 10 variables can be selected
(b) a maximum variable value can be selected
(c) eligible operators can be selected (+ - / * ^).

The program randomly generates an algebraic expression using the selected parameters.

The quizzes mirror their practice counterparts, using the same problem presentation formats. The difference between Practice exercises and Quizzes is that quizzes are TIMED and GRADED.

Furthermore, high scores are saved to promote competition.

II. SOLVE MODE

The method used to solve order of operations problems involve two algorithms.

1) The first algorithm (SHUNTING YARD - credit: Dijkstra) translates our equation from "infix" notation, where operators fall in between the operands they're acting upon (e.g. 1 + 2 * 3) to "postfix" notation, where all operators follow their operands (e.g. 1 2 3 * +). This is important because when the machine is reading expressions in this format, it will never encounter an operator before the operands it’s acting on, which means it won’t have to go back and forth between operator and operands.

2) The second algorithm (POSTFIXEVAL) evaluates the Postfixed expression to produce the solution.

Both algorithms utilize arrays known as stacks and queues.

A stack is analogous to a stack of dishes. You add a new dish to the top of the stack, and all dishes currently in the stack are "pushed" down one level. Retrieving values from the stack ("popping") retrieves the topmost value, just like you would taking a dish from the stack. In computer parlance, this is known as Last In, First Out or LIFO.

A queue on the other hand resembles group of people waiting in line for an event. As new people join the group, they take their place at the end of the line. Removing a value from a queue is analogous to the first person in line leaving the line to enter the event and others in line moving forward one place. In computer parlance, this is known as First In, First Out or FIFO.

An Order of Operation Precedence Table is established, using the familiar PEMDAS rule:

1) Parenthesis

2) Exponent

3) Multiplication/Division (equal precedence)

4) Addition/Subtraction (equal precedence).

Because parenthetical content is the highest priority, the program first scans for a parenthesis pair. If found, its content is translated with SHUNTING YARD, then solved with POSTFIXEVAL. The resulting value is substituted back into the original expression, taking the place of the parenthetical information, leaving a parenthesis-free expression, which can then be solved using the same SHUNTING YARD/POSTFIXEVAL methodology as before.

The SHUNTING YARD Steps:

1) Parse expression left to right, element by element.

2) If element is

2.1) an OPERAND, immediately add to Queue; parse next element

2.2) an OPERATOR, then:

2.2.1) if stack is empty, immediately push operator onto the stack; parse next element

2.2.2) if stack is not empty, then:

2.2.2.1) if incoming operator has HIGHER precedence than top of stack, push incoming operator onto stack; parse next element

2.2.2.2) if incoming operator has EQUAL precedence to top of stack, pop top of stack and add that operator to the queue, then push incoming operator onto stack; parse next element

2.2.2.3) if incoming operator has LOWER precedence to top of stack, pop top of stack and add that operator to the queue then compare incoming operator with new top of stack and follow rule 2.2.1 above.

3. Once the entire expression has been parsed, pop all remaining operators on the stack and add to queue.

Example: 1+2*3^4

Rule 2.1: QUEUE [1] STACK [empty]

Rule 2.2.1 QUEUE [1]
STACK [+]

Rule 2.1: QUEUE [1, 2]
STACK [+]

Rule 2.2.2.1 QUEUE [1, 2]
STACK [*]
[+]

Rule 2.1 QUEUE [1, 2, 3]
STACK [*]
[+]

Rule 2.2.2.1 QUEUE [1, 2, 3]
STACK [^]
[*]
[+]

Rule 2.1 QUEUE [1, 2, 3, 4]
STACK [^]
[*]
[+]

Rule 3 QUEUE [1, 2, 3, 4,^,*,+]
STACK [empty]

Final POSTFIX EXPRESSION: 1 2 3 4 ^ * +

Note that in the case of parentheses, the subexpression withing the parenthesis is isolated and evaluated using SHUNT YARD. That expression is evaluated (see below) and its result substituted for the parenthetical subexpression. This yields a parenthesis-free string which is then translated using the SHUNT YARD algorithm.

POSTFIXEVAL Steps:

Start with empty stack

1) Parse first token in PostFix expression left to right, element by element.

2) If element is

2.1) an OPERAND, immediately push onto stack

2.2) an OPERATOR, then pop off two operands from stack and save to temporary variables.

2.2.1) evaluate the expression:

```
EVAL = Temp1 [OPERATOR] Temp2
```

2.2.2) push EVAL to top of stack

3. Repeat until expression is empty. Solution will be lone stack value.

Example: 1 2 3 4 ^ * + (Postfix expression from previous example)

EXPRESSION [1 2 3 4 ^ * +]

STACK [empty]

Rule 2.1: EXPRESSION [2 3 4 ^ * +]

STACK [1]

Rule 2.1: EXPRESSION [3 4 ^ * +]

STACK [2]

STACK [1]

Rule 2.1: EXPRESSION [4 ^ * +]

STACK [3]

STACK [2]

STACK [1]

Rule 2.1: EXPRESSION [^ * +]

STACK [4]

STACK [3]

STACK [2]

STACK [1]

Rule 2.2: Operator = "^"

Temp2 = pop 4

Temp1 = pop 3

Rule 2.2.1 EVAL = Temp1 Operator Temp2 => 3 ^ 4 = 81

Rule 2.2.2 STACK [81]

STACK [2]

STACK [1]

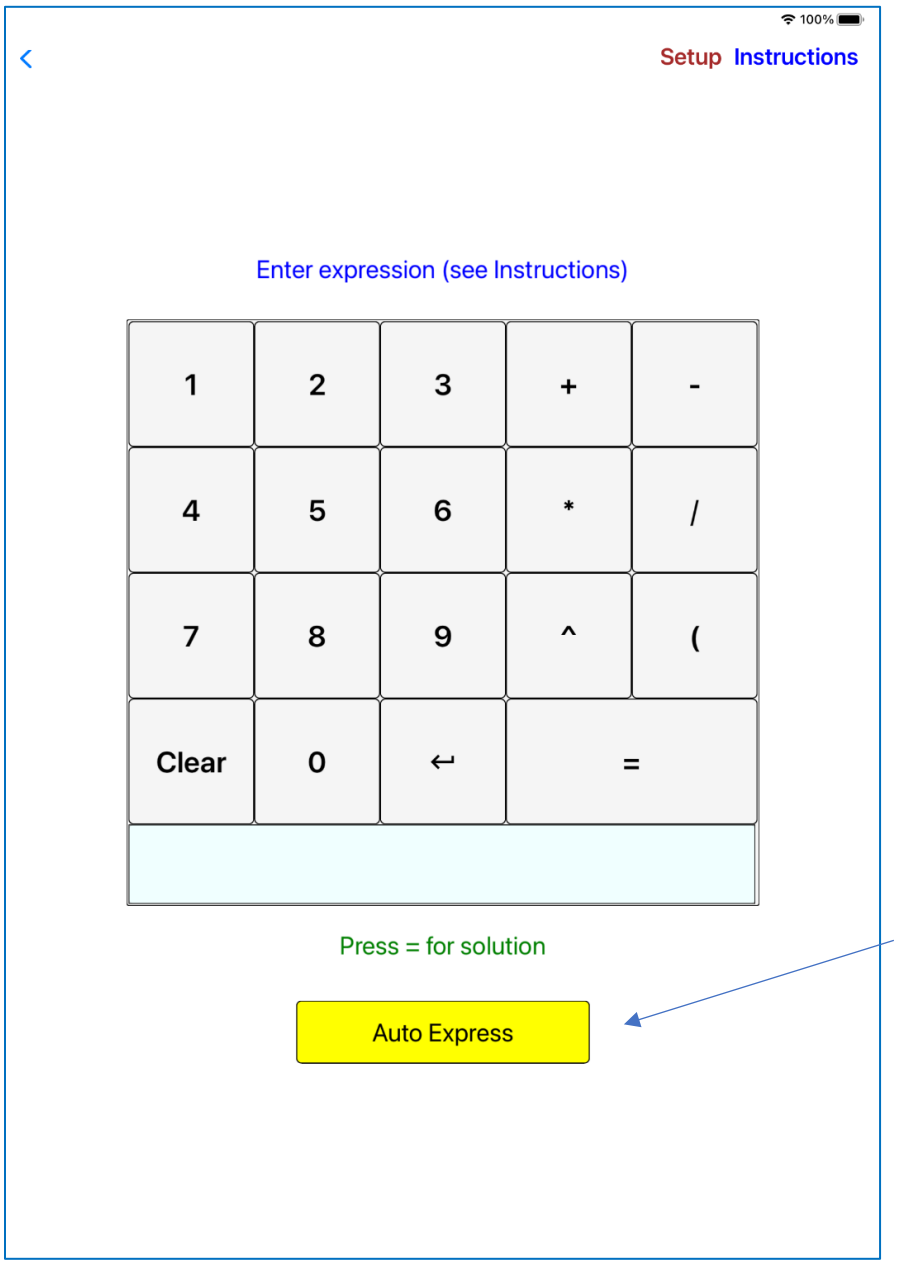
```
Rule 2.1: EXPRESSION [* +]

Rule 2.2: Operator = "*"
          Temp2 = pop 81
          Temp1 = pop 2

Rule 2.2.1     EVAL = Temp1 Operator Temp2 => 2 * 81 = 162

Rule 2.2.2     STACK [162]
          STACK [1]

Rule 2.1: EXPRESSION [+]

Rule 2.2: Operator = "+"
          Temp2 = pop 162
          Temp1 = pop 1

Rule 2.2.1     EVAL = 1 + 162 = 163

Rule 2.2.2     STACK [163]

Rule 3.   EXPRESSION [empty]
          SOLUTION = pop 163
```

III. PRACTICE MODE

The opening screen features a custom calculator containing only the keys needed to manually enter a polynomial expression to be evaluated.

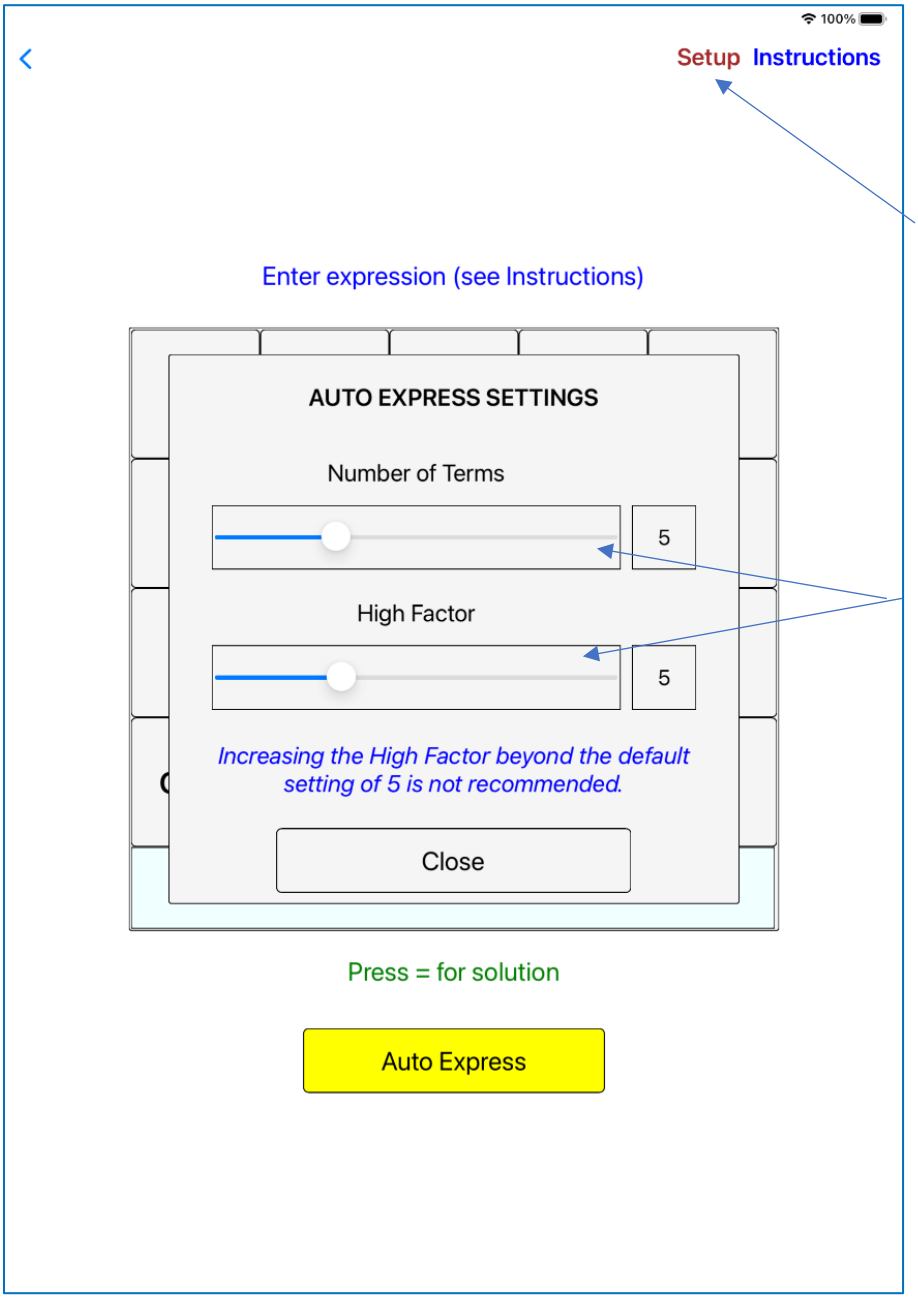
Expressions can be entered freehand-style using the keyboard, or a random polynomial expression can be auto-generated.

The opening Practice Mode Screen:



There is a maximum limit of ten variables, but no limit to their magnitude in the practice mode.

For an auto-generated entry, press the [Auto Express] button to generate a random polynomial expression.



The Auto Express option has a separate Setup page

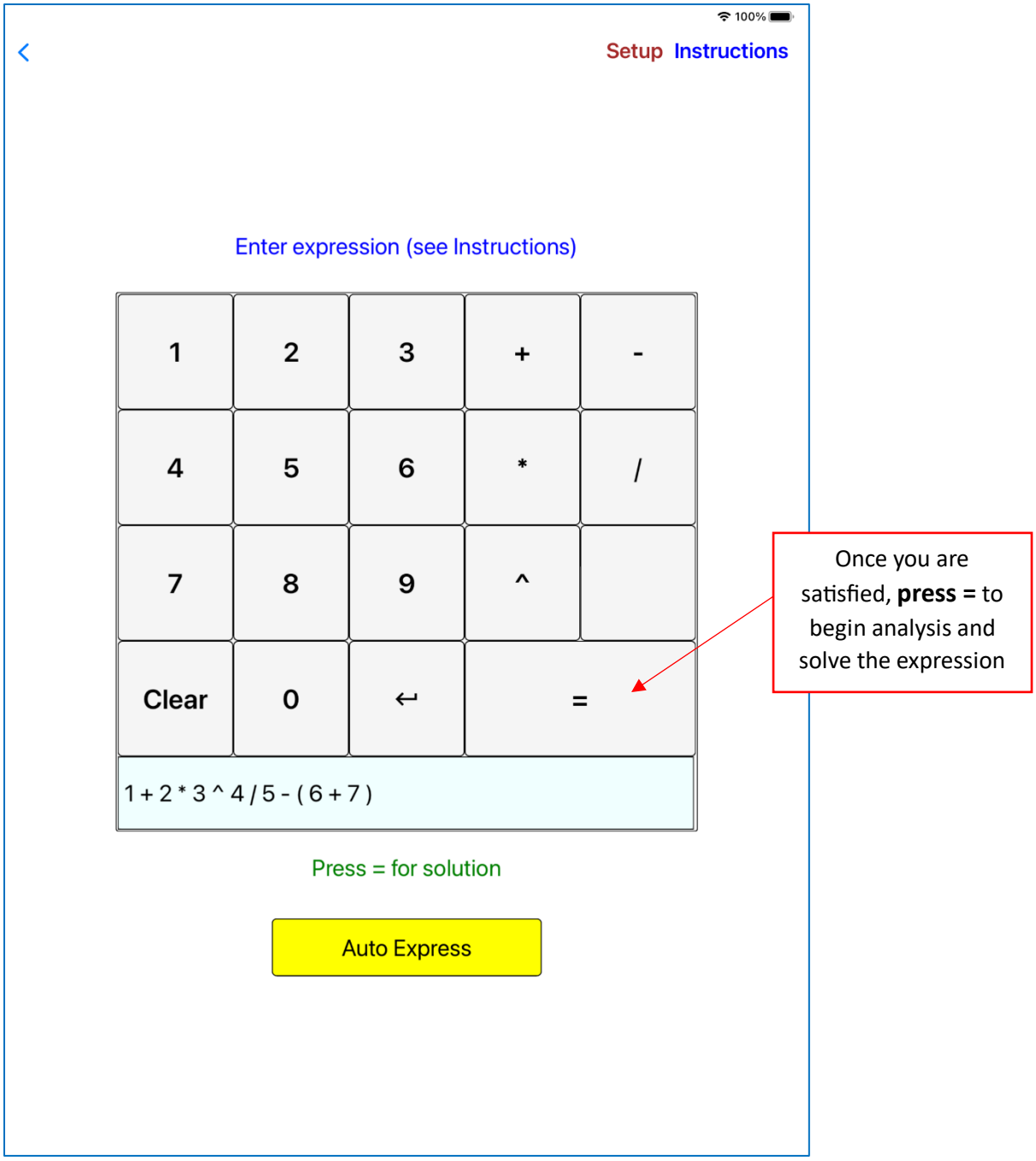
The sliders are used to set the number of terms in the polynomial and the highest factor in the expression.

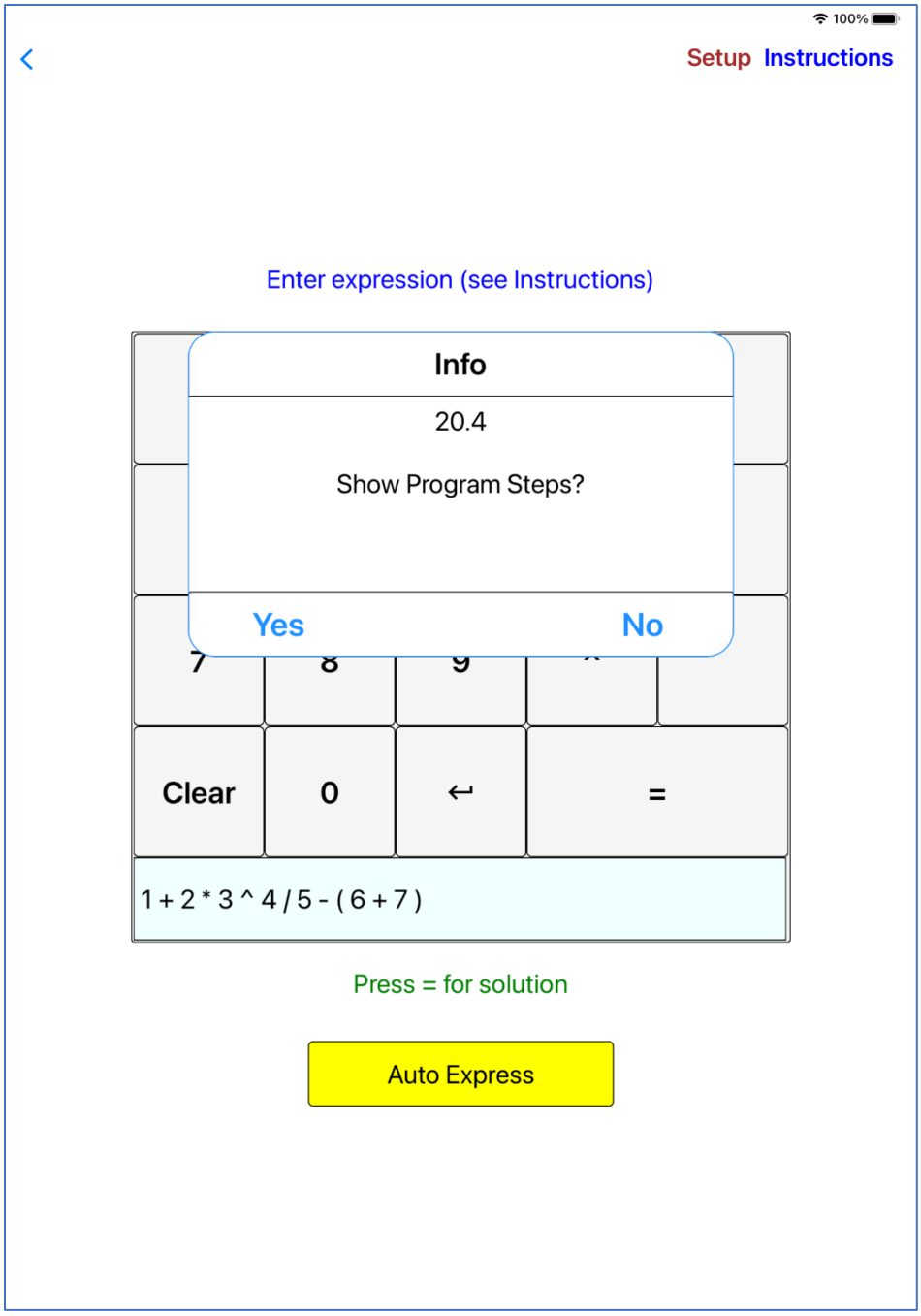
The auto-generated expression will have randomly selected factors and operators.

Because of the possibility of an exponent (power) operator being randomly selected, it is not advisable to increase the High Factor beyond 5.

Freehand Example:

Using the keyboard, enter a string of numbers and operators. In this example, all the available operators and one level of parenthesis are entered.



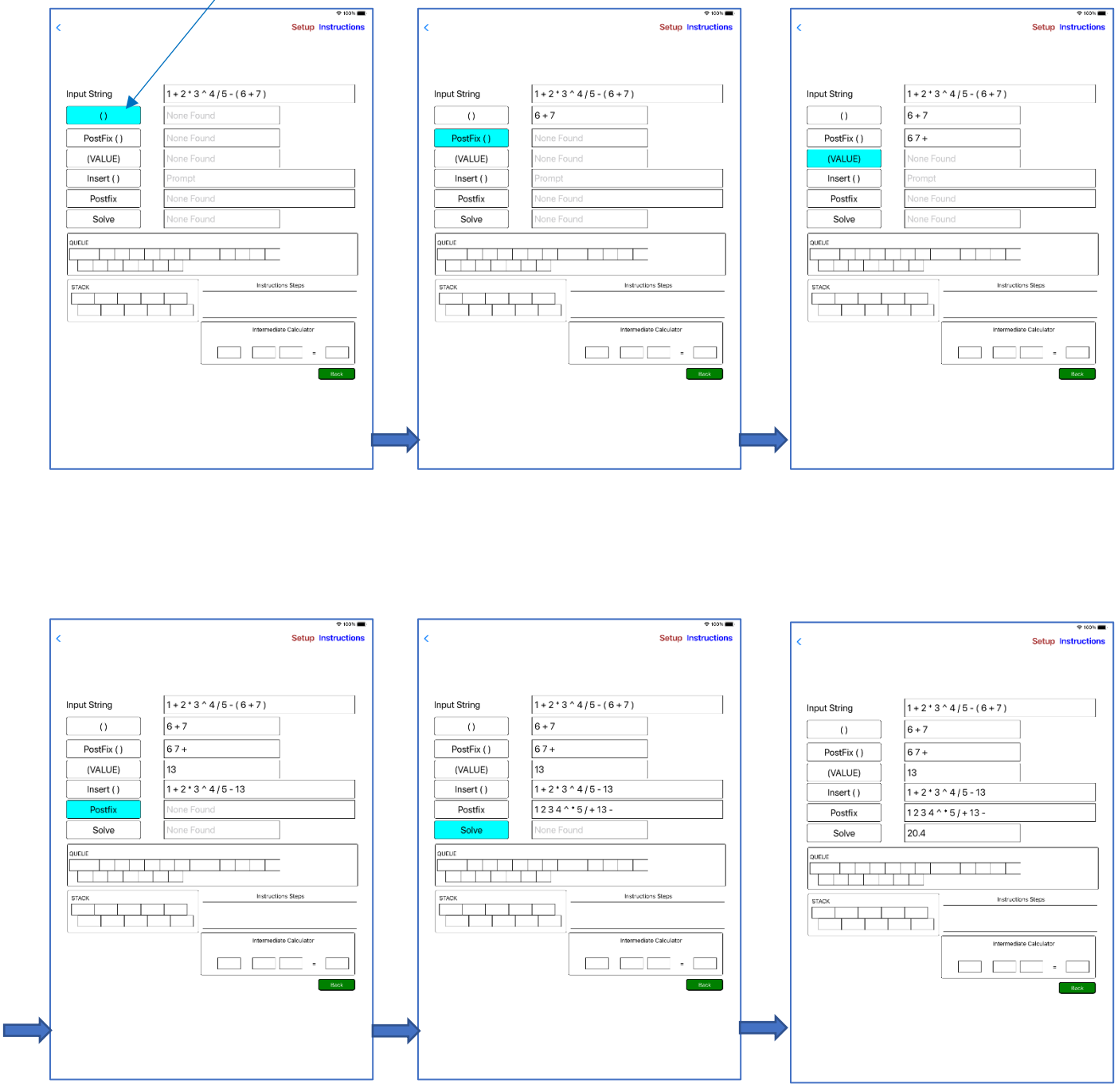


The solution to the expression is displayed in an information box along with an invitation to see a step-by-step analysis of the solution, utilizing the SHUNT YARD algorithm.

(A discussion of this algorithm was previously explained)

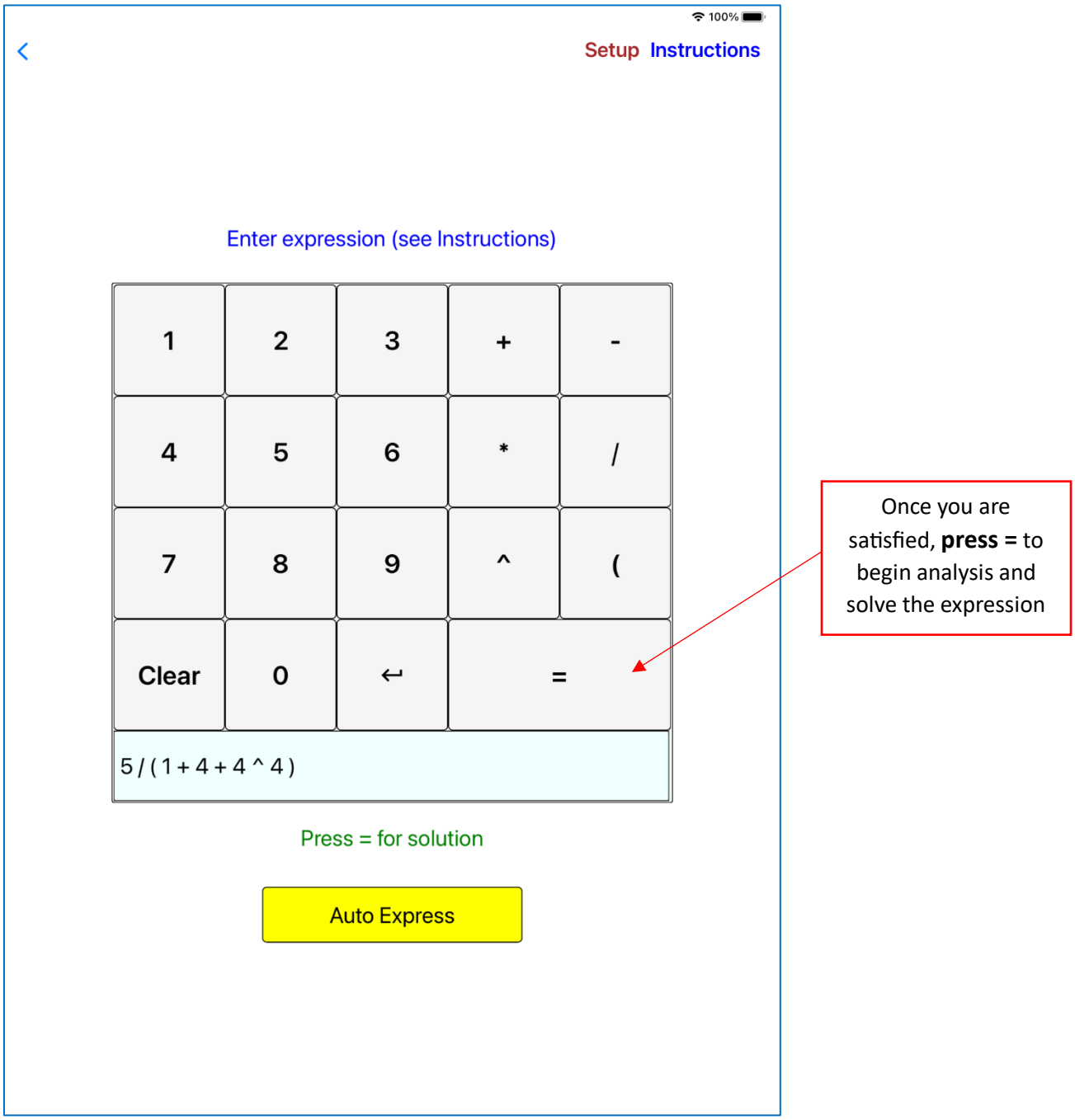
This series of screenshots depict stepping though the Shunt Yard algorithm. Intermediate results are shown with each step and the queue and stack activity merit special attention.

Pressing the **highlighted buttons** advance the algorithm to the next step.



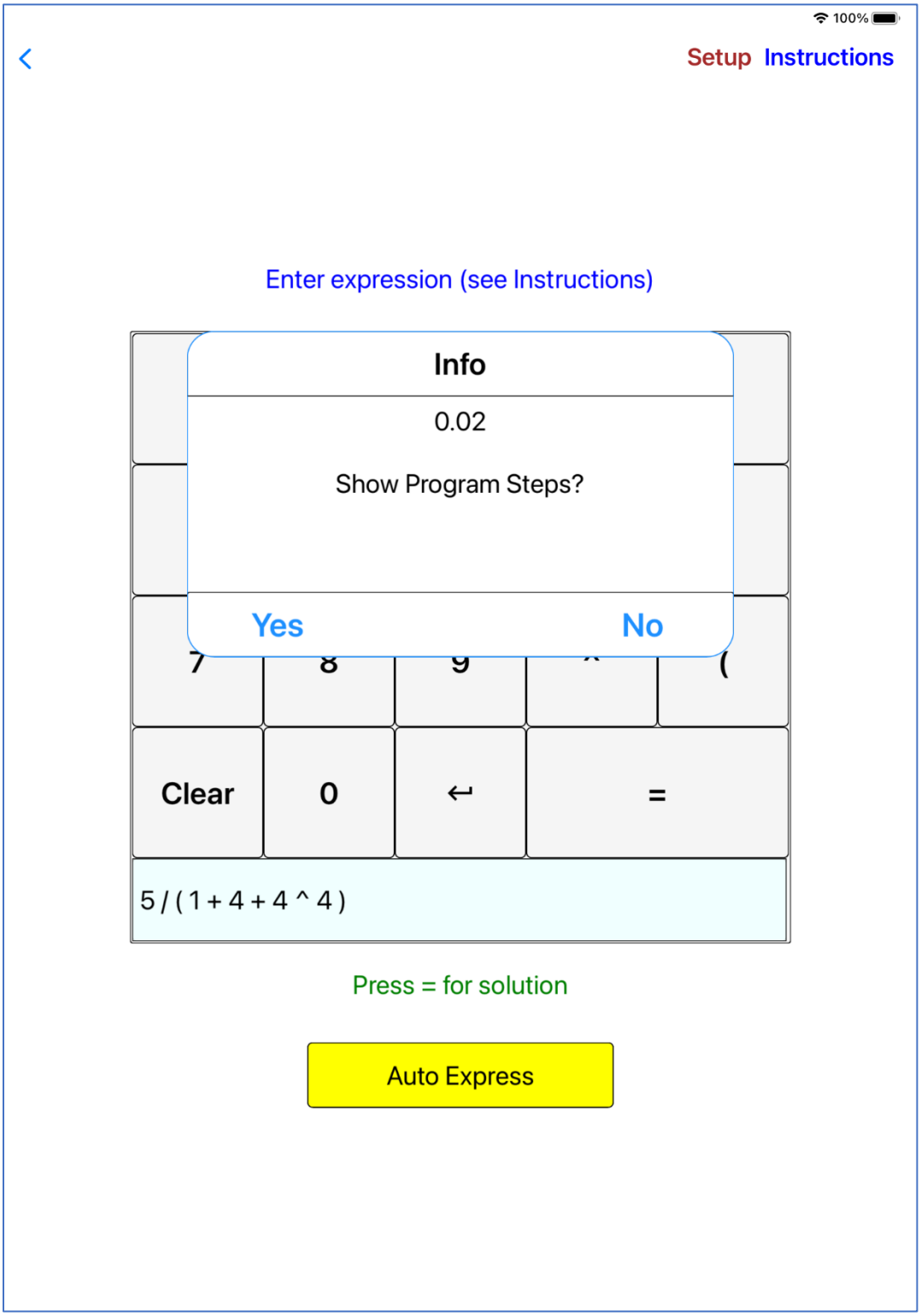
Auto Express Example:

Press the **Auto Express** button.



Repeated presses will generate new random expressions.

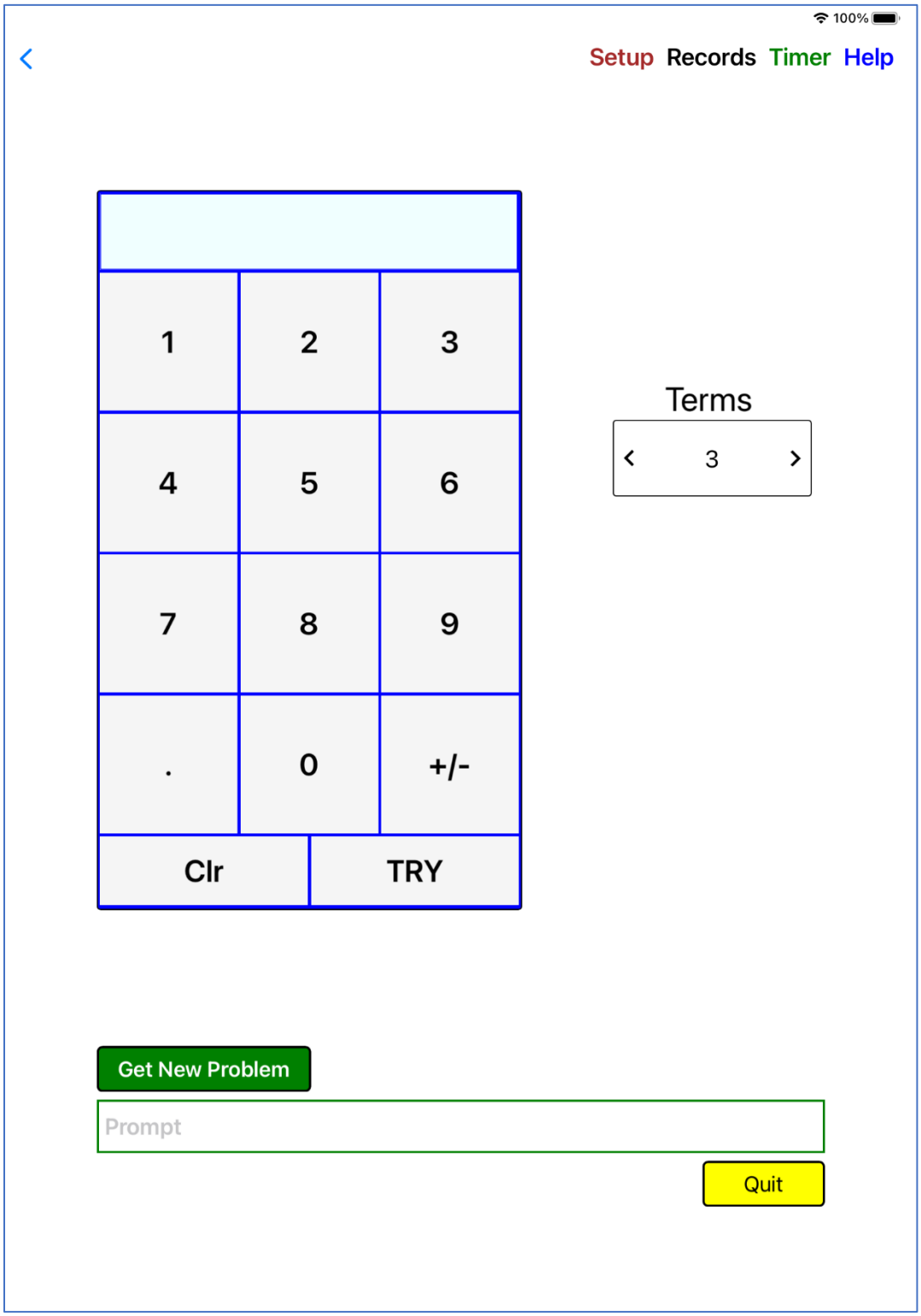
As with the freehand example, the student is given the option of examining the answer by stepping through the algorithm.



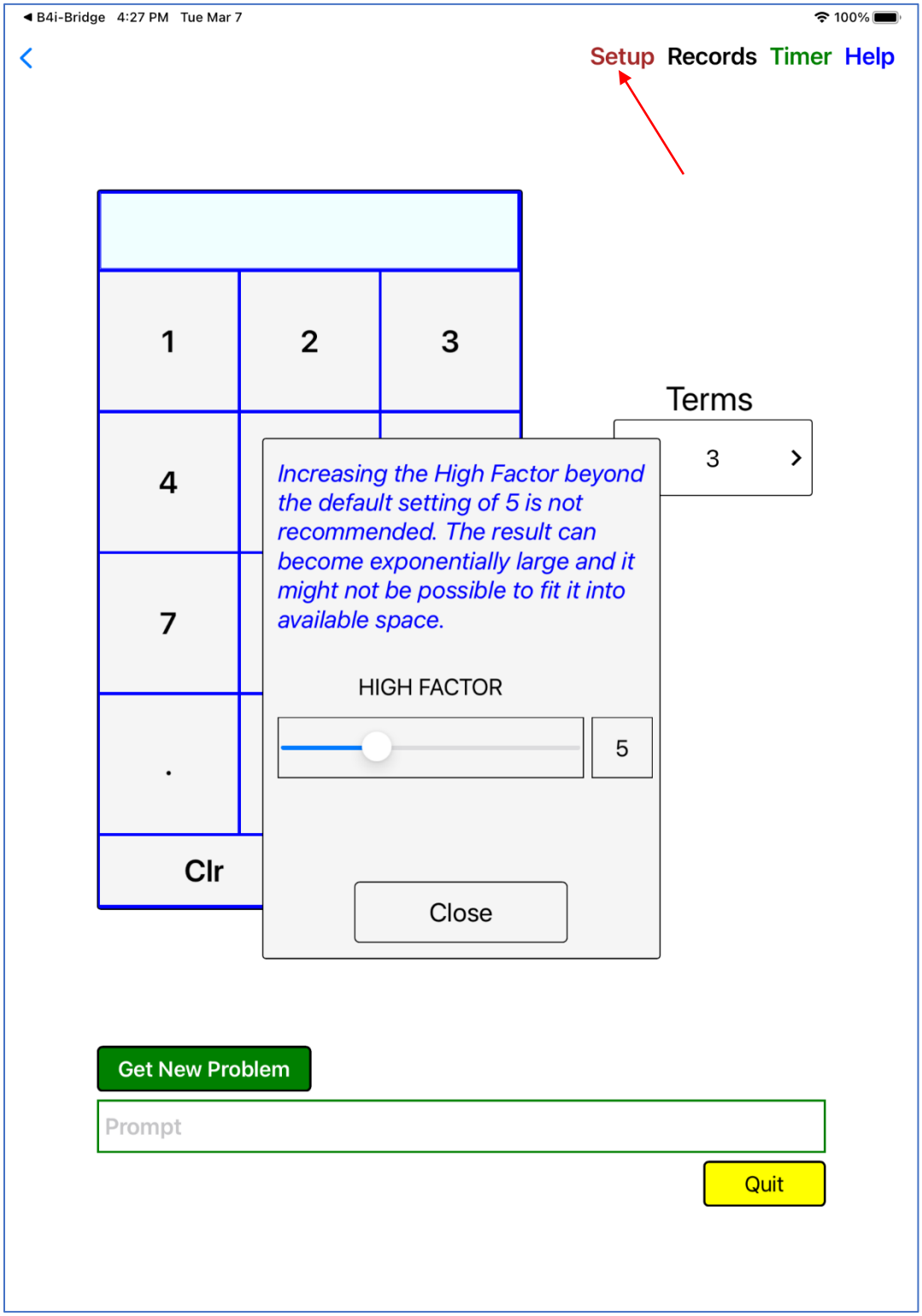
In the event that the solution has a fractional component, the program rounds to two decimal places.

IV. QUIZ MODE

The opening screen:

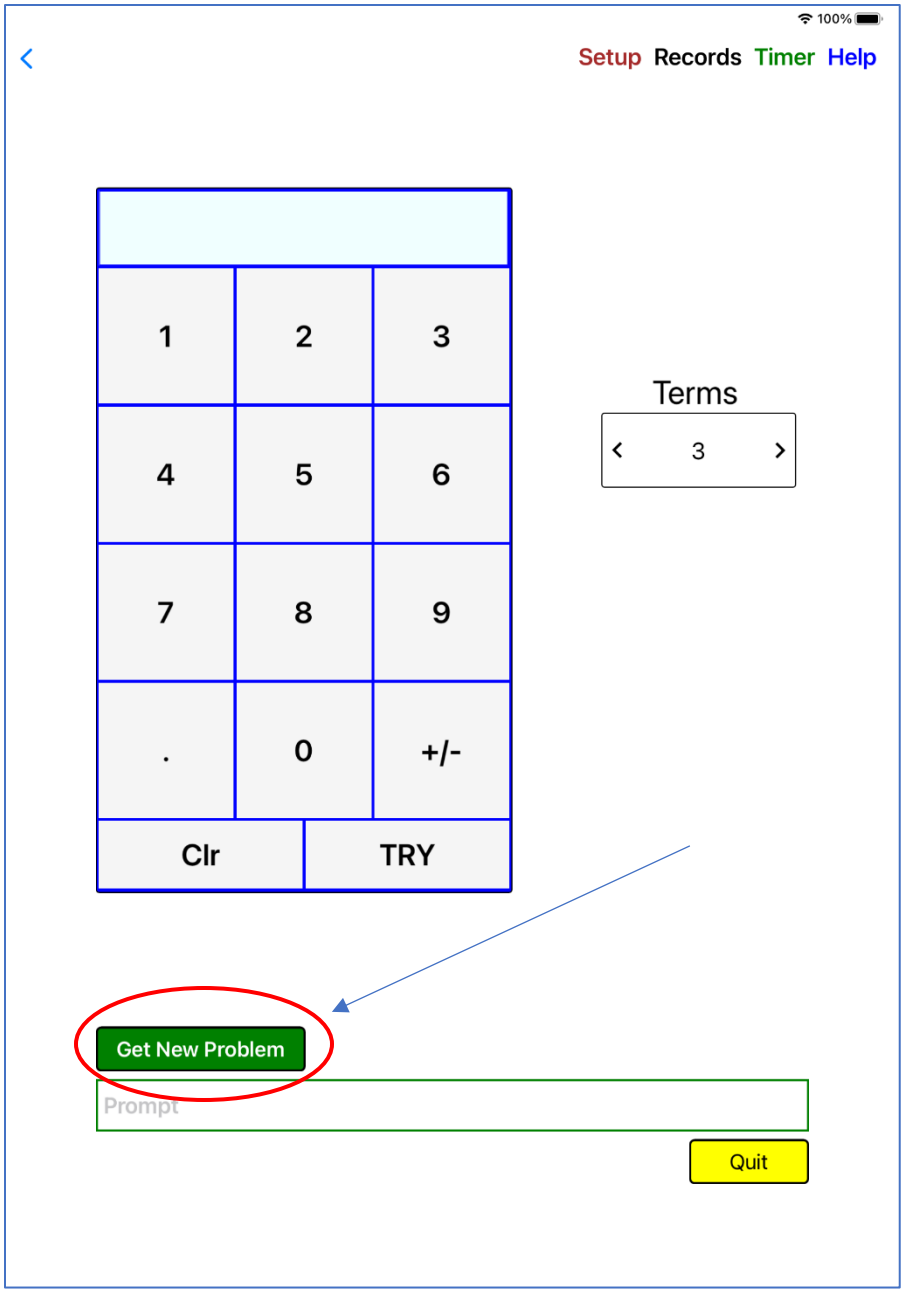


Before starting a quiz, use the Options menu at the top of the screen to set up the only parameter setting for the quiz, the High Factor.



It is advisable to leave the High Factor set at 5 or less. After all, the point of the quiz is to test Order of Operations knowledge, and not to just generate extremely complicated math problems, the answers of which might not fit into available screen space.

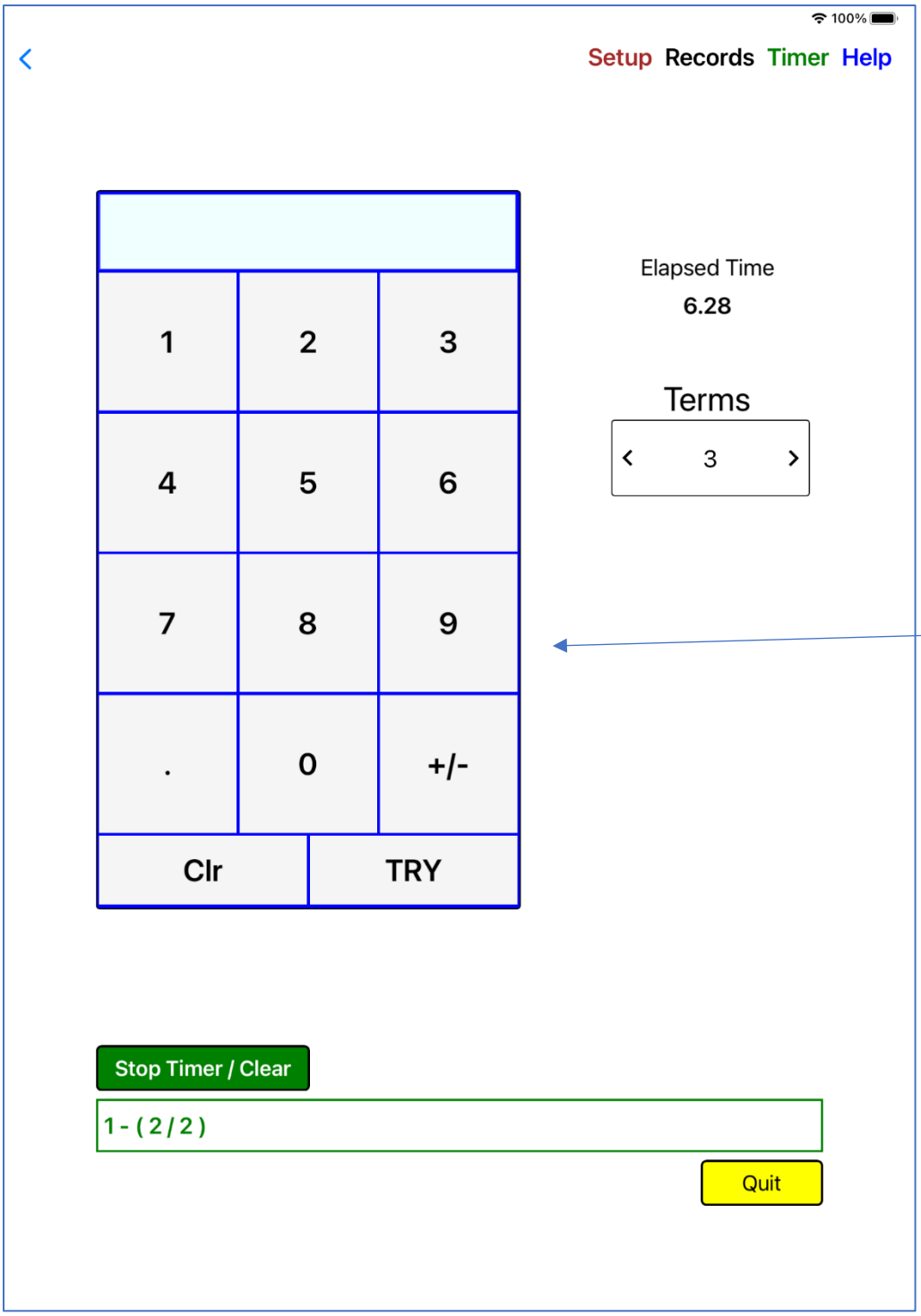
When ready to start the quiz, press Get New Problem button.



A polynomial expression will be auto-generated containing the number of terms selected using the left/right TERMS selector.

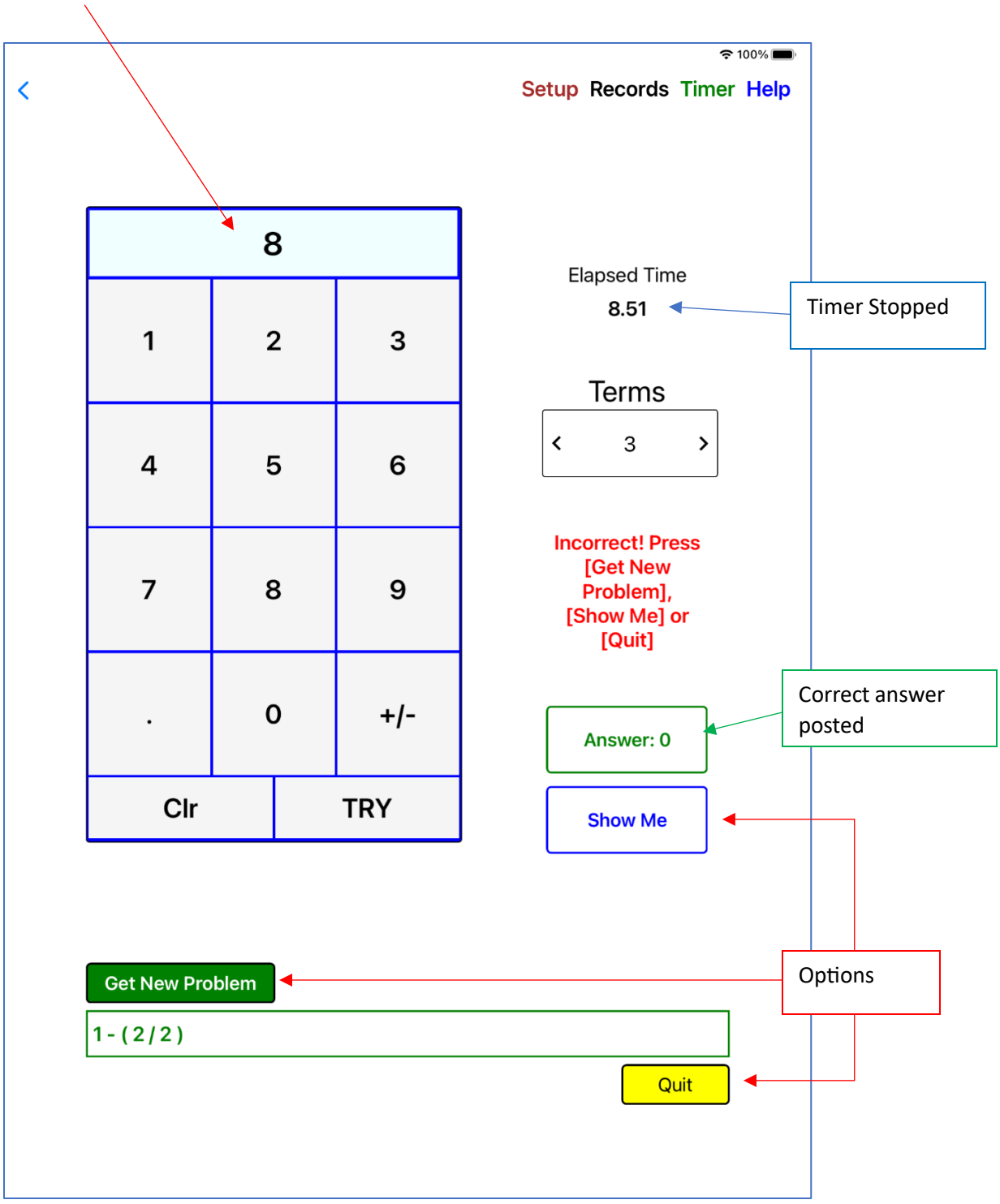
The maximum value for any term is determined by the High Factor setting.

If the Timer status is ON, then pressing the Get New Problem also starts the timer, which will be used to determine if the time needed to find the solution is lower than the time on record for quizzes with the same number of terms.



Step 2. Use the keyboard to enter the value of the expression.

Here, a wrong answer was entered and the TRY button pressed.

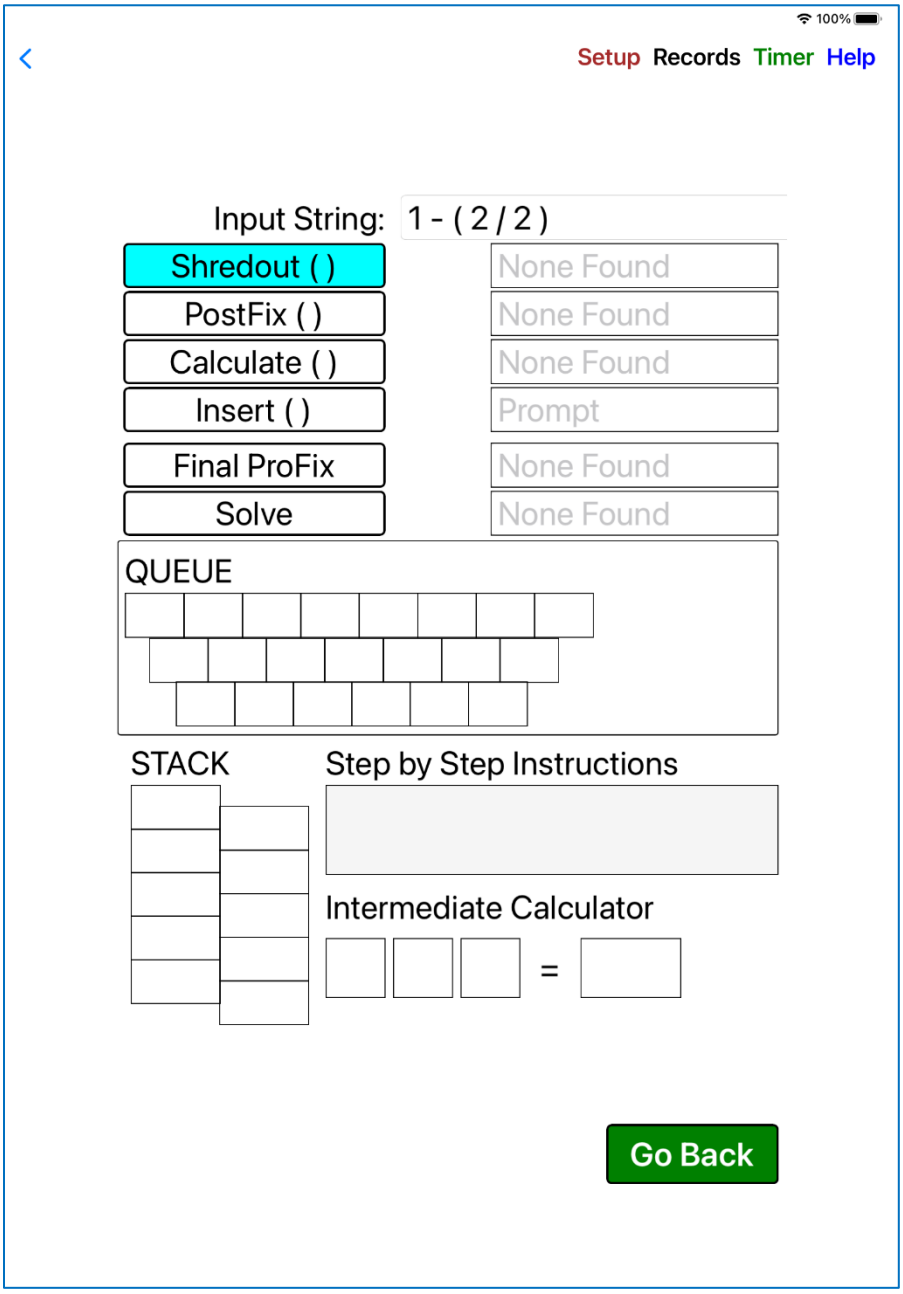


As a result, the timer is stopped, the correct answer is posted and the student has 3 options:

- Get a New Problem
- Show Me
- Quit

The SHOW ME option.

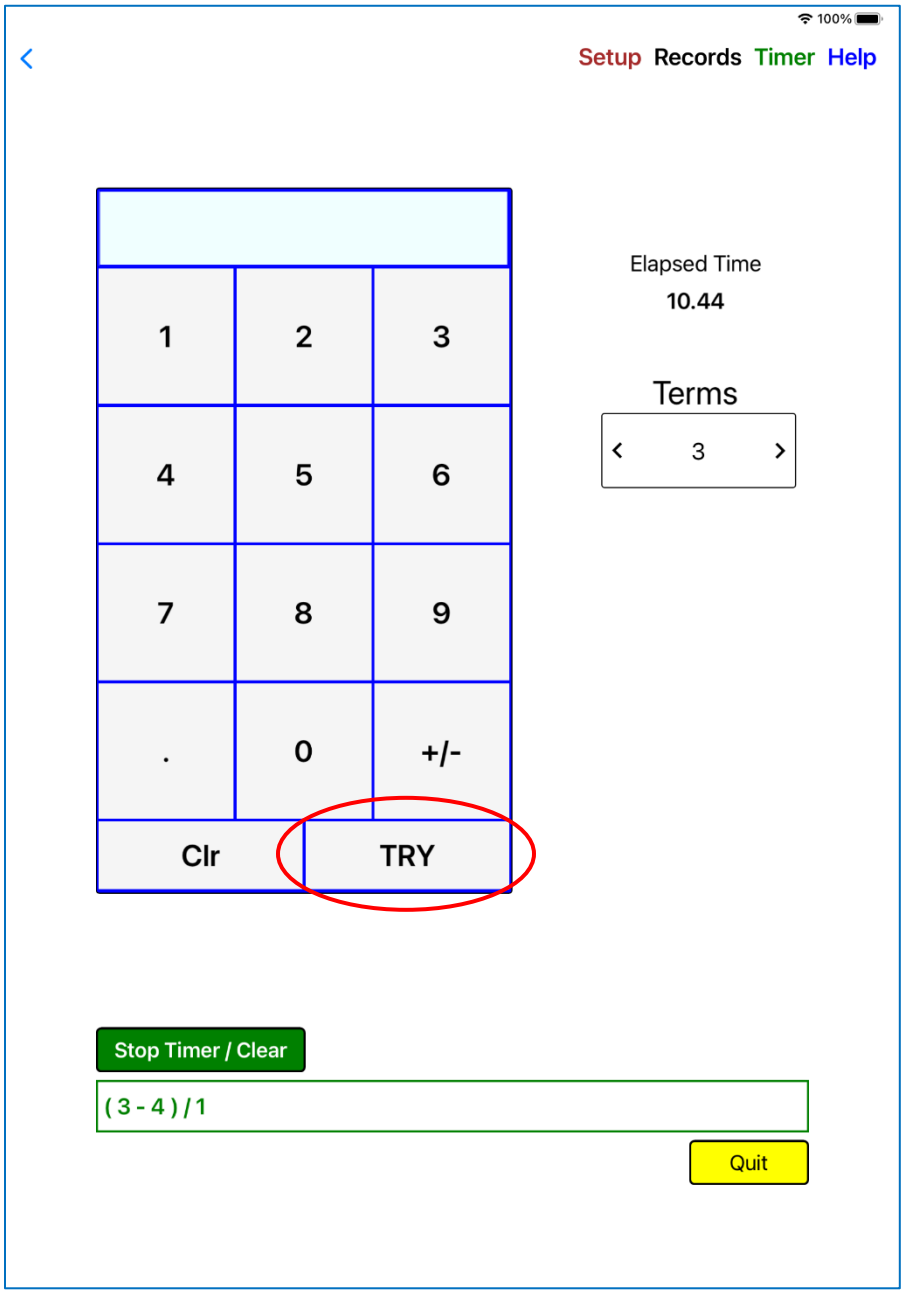
(Refer to Section II for detailed explanation)



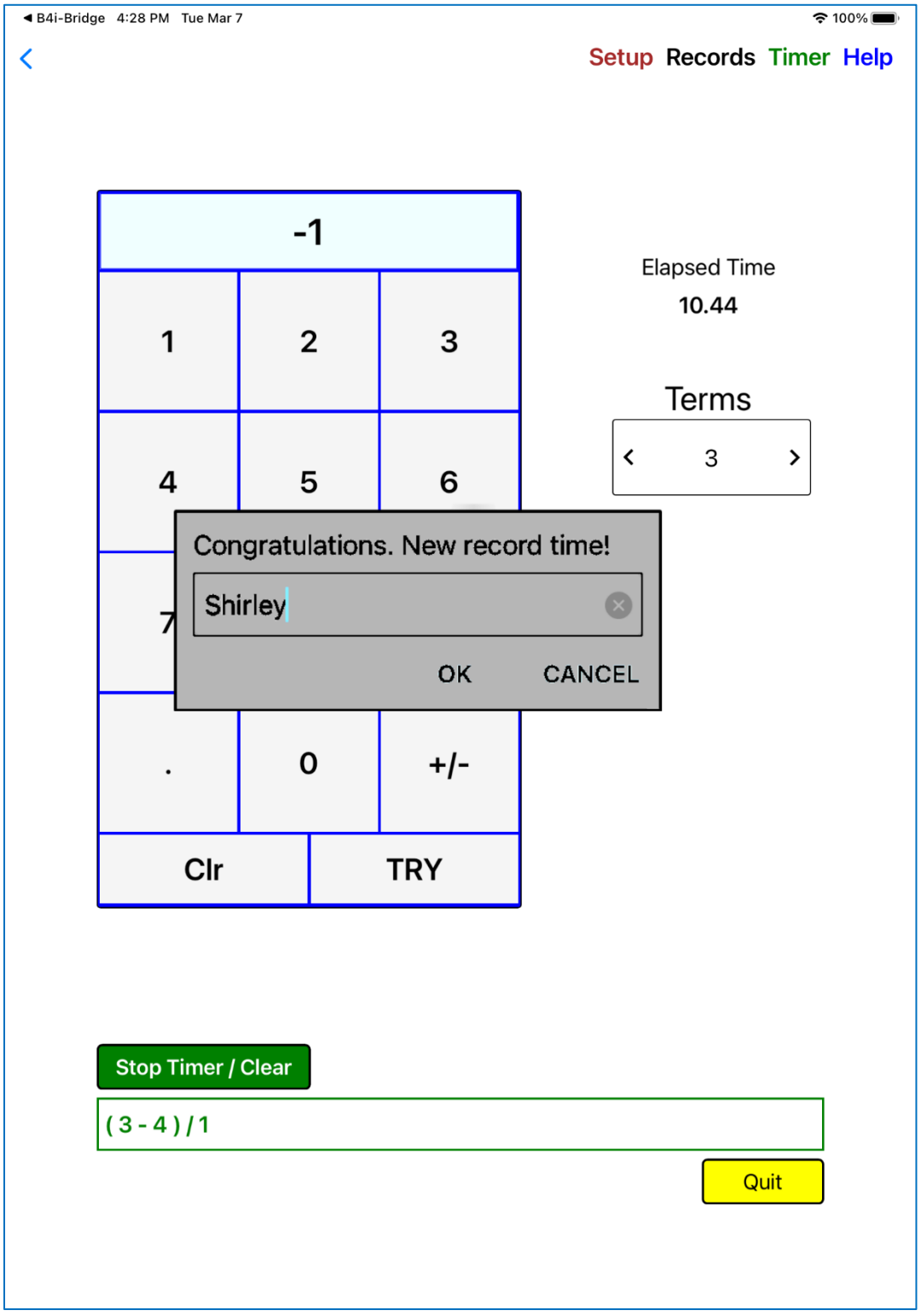
Let's assume that the Get a New Problem button is pressed. This results in the following:

- The answer window is cleared,
- A new expression is presented,
- The timer is reset and
- The keyboard is opened for input.

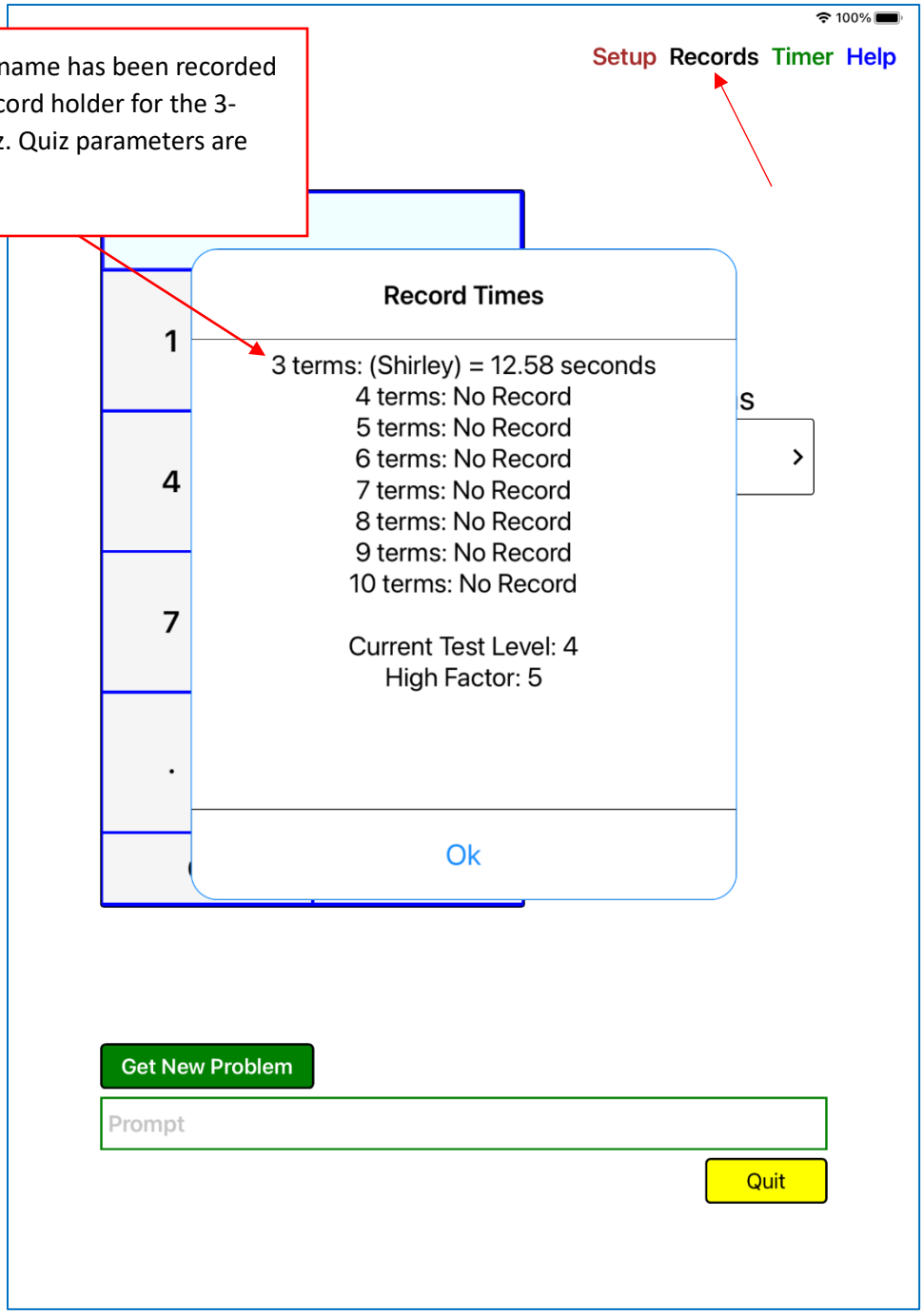
Enter the answer (-1) using the keyboard. Use the +/- key to enter a negative number. Then press the TRY button.



In this example, when a comparison was made to the previous record low time for this expression length (number of terms), the comparison showed that this test time was lower. The student is asked for a name to enter into the record books.

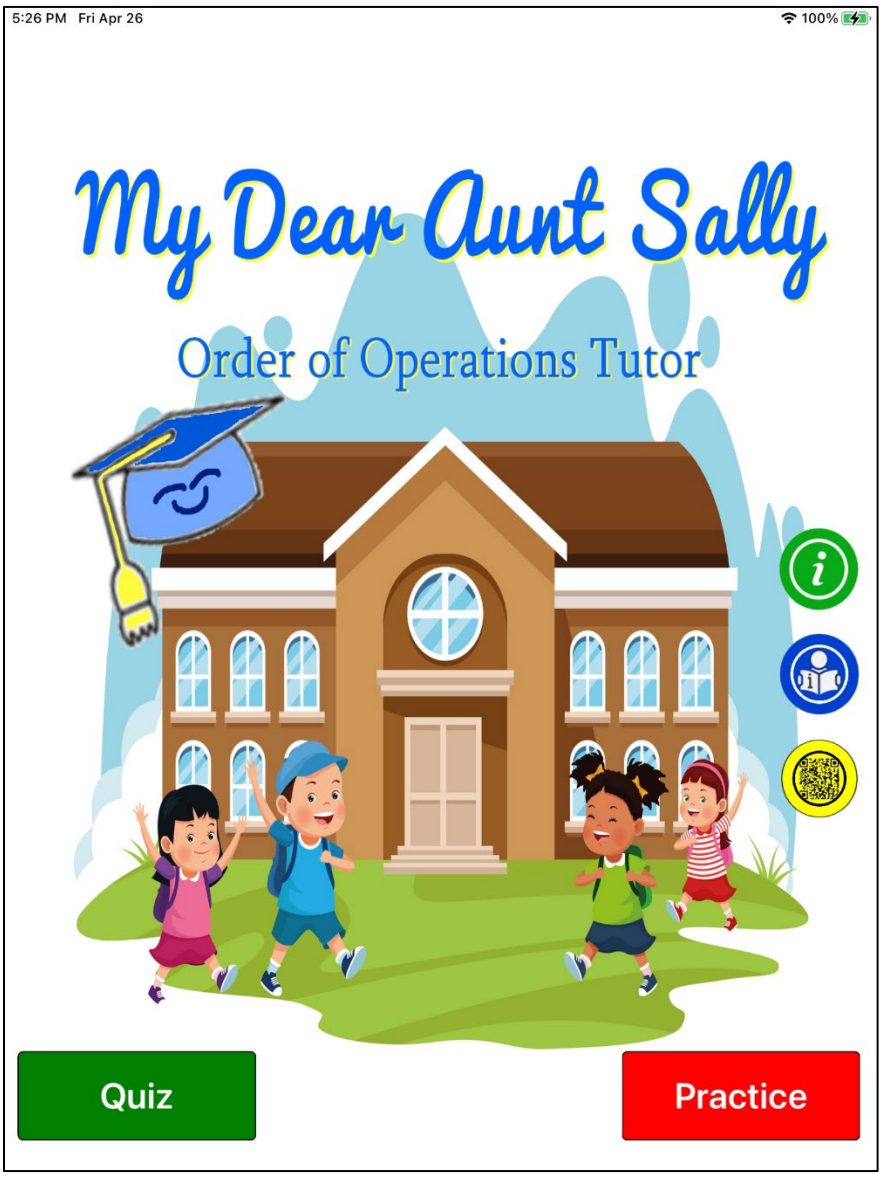


Return to the Quiz home screen and select the Records menu item, then Show Records



Quizzes always consist of a single question. The complexity increases as the number of terms increases, which is what the Record levels are based on.

Menus and Controls



About: Version, copyright. Option to Rate/Review the app

Guide: Print or View the User Guide

QR: QR Code to quickly access the TurboSoftSolutions.Com website

Product Name: My Dear Aunt Sally

Company: TurboSoftSolutions.Com

Programmer: Neil Rohan