Problets Anyone? (problets.org)

Amruth N. Kumar
Ramapo College of New Jersey
amruth@ramapo.edu

What Problets do:

- Present problems
- Grade student’s answer
- Provide instant feedback
- Record student performance
- Provide summary to the instructor

Types of Problems

- “Debug this program”
- “What is printed by this program?”
- “Show the changes to the array after the loop”
- “Evaluate the expression step-by-step”

→

- *Not* multiple-choice problems

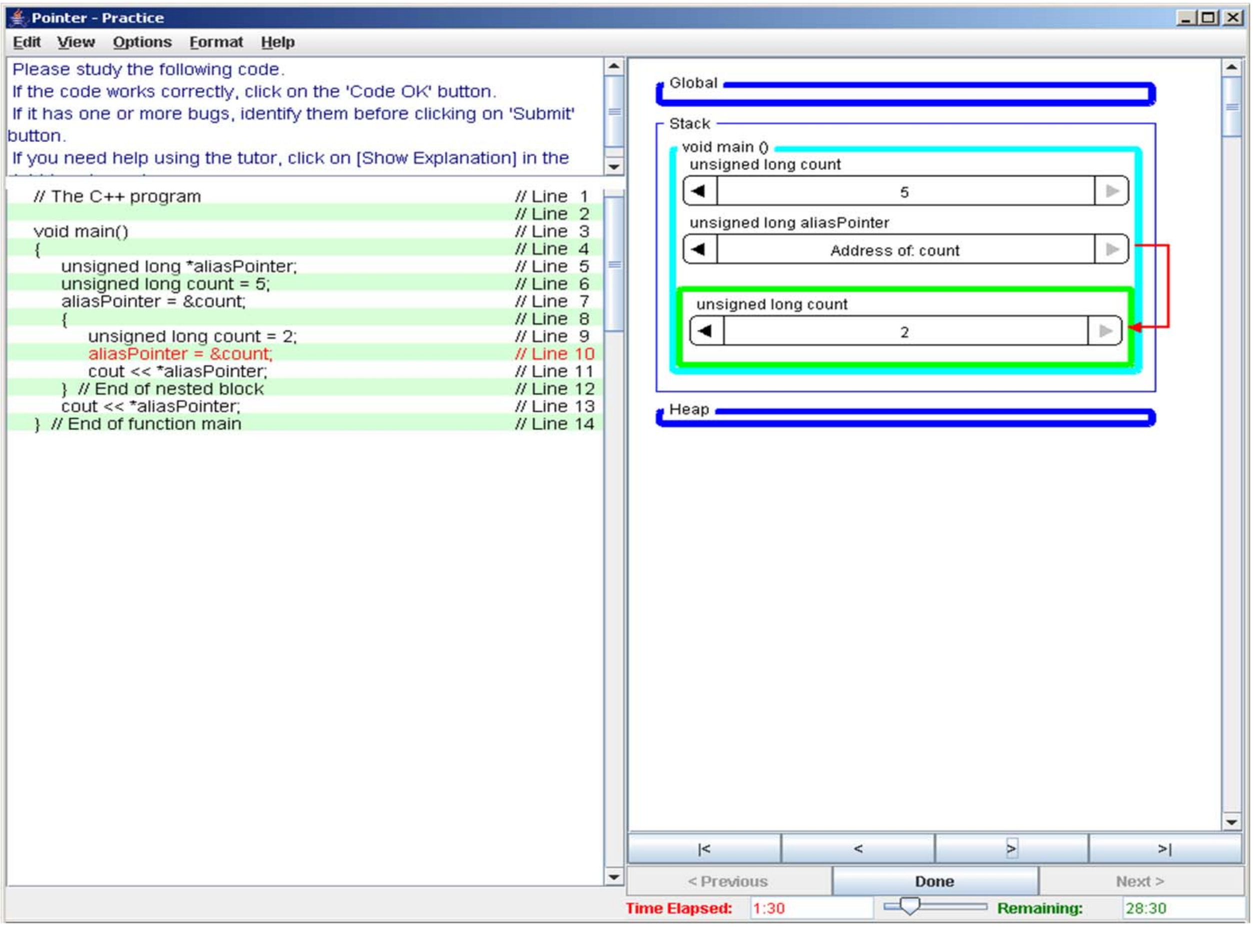
Features of Problem Generation

- *Adapted* to the learning needs of the student
  - Maximizes learning in minimal time
- No two problems are alike
  - Deters cheating, plagiarism
- Problems are presented ad-infinitum
  - Supports as many practice exercises as necessary

Types of Feedback

- Error-Flagging
- Whether the answer is correct
- *Step-by-step explanation* of the correct answer
- Graphic visualization

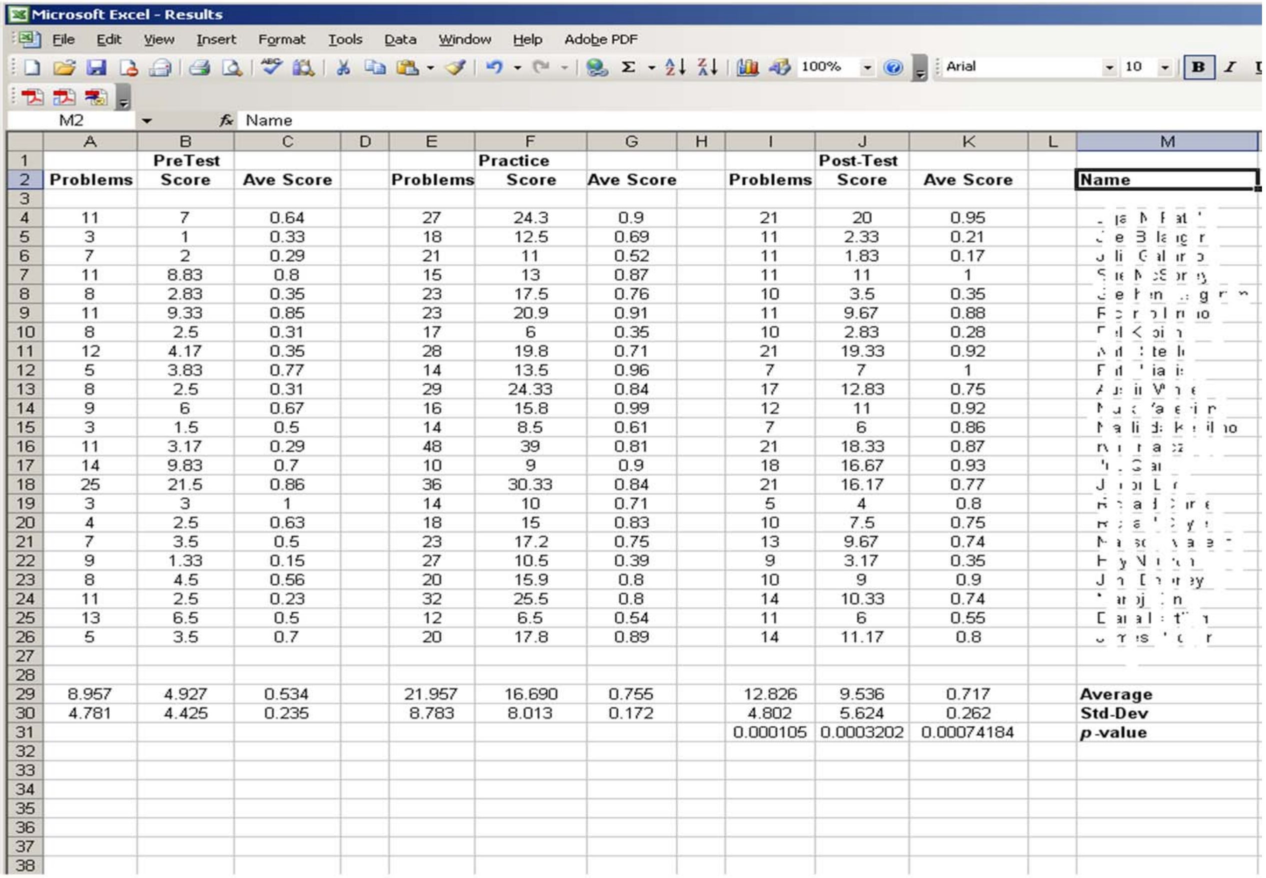
Visualization of Data flow



Please study the following code.
If the code works correctly, click on the 'Code OK' button.
If it has one or more bugs, identify them before clicking on 'Submit' button.
If you need help using the tutor, click on [Show Explanation] in the

```
  // The C++ program                                  // Line  1
                                                      // Line  2
  void main()                                         // Line  3
  {                                                   // Line  4
      unsigned long *aliasPointer;                    // Line  5
      unsigned long count = 5;                        // Line  6
      aliasPointer = &count;                          // Line  7
      {                                               // Line  8
          unsigned long count = 2;                    // Line  9
          aliasPointer = &count;                      // Line 10
          cout << *aliasPointer;                      // Line 11
      }  // End of nested block                       // Line 12
      cout << *aliasPointer;                          // Line 13
  }  // End of function main                          // Line 14
```

Instructor’s Summary


Microsoft Excel - Results
File Edit View Insert Format Tools Data Window Help Adobe PDF
100%
Arial
10
M2
Name

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		PreTest				Practice				Post-Test			
2	Problems	Score	Ave Score		Problems	Score	Ave Score		Problems	Score	Ave Score		Name
3													
4	11	7	0.64		27	24.3	0.9		21	20	0.95		[redacted]
5	3	1	0.33		18	12.5	0.69		11	2.33	0.21		[redacted]
6	7	2	0.29		21	11	0.52		11	1.83	0.17		[redacted]
7	11	8.83	0.8		15	13	0.87		11	11	1		[redacted]
8	8	2.83	0.35		23	17.5	0.76		10	3.5	0.35		[redacted]
9	11	9.33	0.85		23	20.9	0.91		11	9.67	0.88		[redacted]
10	8	2.5	0.31		17	6	0.35		10	2.83	0.28		[redacted]
11	12	4.17	0.35		28	19.8	0.71		21	19.33	0.92		[redacted]
12	5	3.83	0.77		14	13.5	0.96		7	7	1		[redacted]
13	8	2.5	0.31		29	24.33	0.84		17	12.83	0.75		[redacted]
14	9	6	0.67		16	15.8	0.99		12	11	0.92		[redacted]
15	3	1.5	0.5		14	8.5	0.61		7	6	0.86		[redacted]
16	11	3.17	0.29		48	39	0.81		21	18.33	0.87		[redacted]
17	14	9.83	0.7		10	9	0.9		18	16.67	0.93		[redacted]
18	25	21.5	0.86		36	30.33	0.84		21	16.17	0.77		[redacted]
19	3	3	1		14	10	0.71		5	4	0.8		[redacted]
20	4	2.5	0.63		18	15	0.83		10	7.5	0.75		[redacted]
21	7	3.5	0.5		23	17.2	0.75		13	9.67	0.74		[redacted]
22	9	1.33	0.15		27	10.5	0.39		9	3.17	0.35		[redacted]
23	8	4.5	0.56		20	15.9	0.8		10	9	0.9		[redacted]
24	11	2.5	0.23		32	25.5	0.8		14	10.33	0.74		[redacted]
25	13	6.5	0.5		12	6.5	0.54		11	6	0.55		[redacted]
26	5	3.5	0.7		20	17.8	0.89		14	11.17	0.8		[redacted]
27													
28													
29	8.957	4.927	0.534		21.957	16.690	0.755		12.826	9.536	0.717		**Average**
30	4.781	4.425	0.235		8.783	8.013	0.172		4.802	5.624	0.262		**Std-Dev**
31									0.000105	0.0003202	0.00074184		***p*-value**
32													
33													
34													
35													
36													
37													
38													

Applications of Problets

- For problem-solving exercises in closed lab or after-class assignment
  - Provides instant feedback
  - Promotes active, self-paced learning
- For in-class testing
  - Automatically grades and logs scores
  - Deters plagiarism/cheating

Topics (C++/Java/C/C#)

- Expression evaluation (Arithmetic, Relational, Logical, Assignment, Bitwise)
- Selection statements (if, if-else, switch)
- Loops (while, for, do-while, break/continue)
- Functions, Recursion
- One-dimensional arrays
- Classes (Access)
- Pointers in C++

Topics and Problems

Topic	Sub-Topic	No. Problem Templates
Expressions	Arithmetic	192
	Relational	268
	Logical	280
	Assignment	255
	Bitwise	303
Selection	If/if-else	165
	switch	147
Loops	while	201
	for	213
	do-while	125
	break, continue	139
Functions	Debugging	117
	Tracing	95
	Recursion	68
Arrays	1-D	172
Classes	Access	128

Logistics

- Instructor requests a problet for a specific language:
  - Free for educational use
- Instructor assigns the problet in a course:
  - Announces the URL
- Students use the problet online:
  - Takes 35-45 minutes
  - Can be used without instructor supervision
- Instructor requests summary

Repeat Adopters

No. of Semesters	No. of Adopters
2	45
3	26
4	10
5	4
6	5
7	2
8+	10

Some Key Findings

- Helps improve learning [SIGCSE 05]
- Helps students learn to write code [ITiCSE 13]
- Helps improve the self-confidence of female students [SIGCSE 08]

Contact Information

Additional information at:

www.problets.org

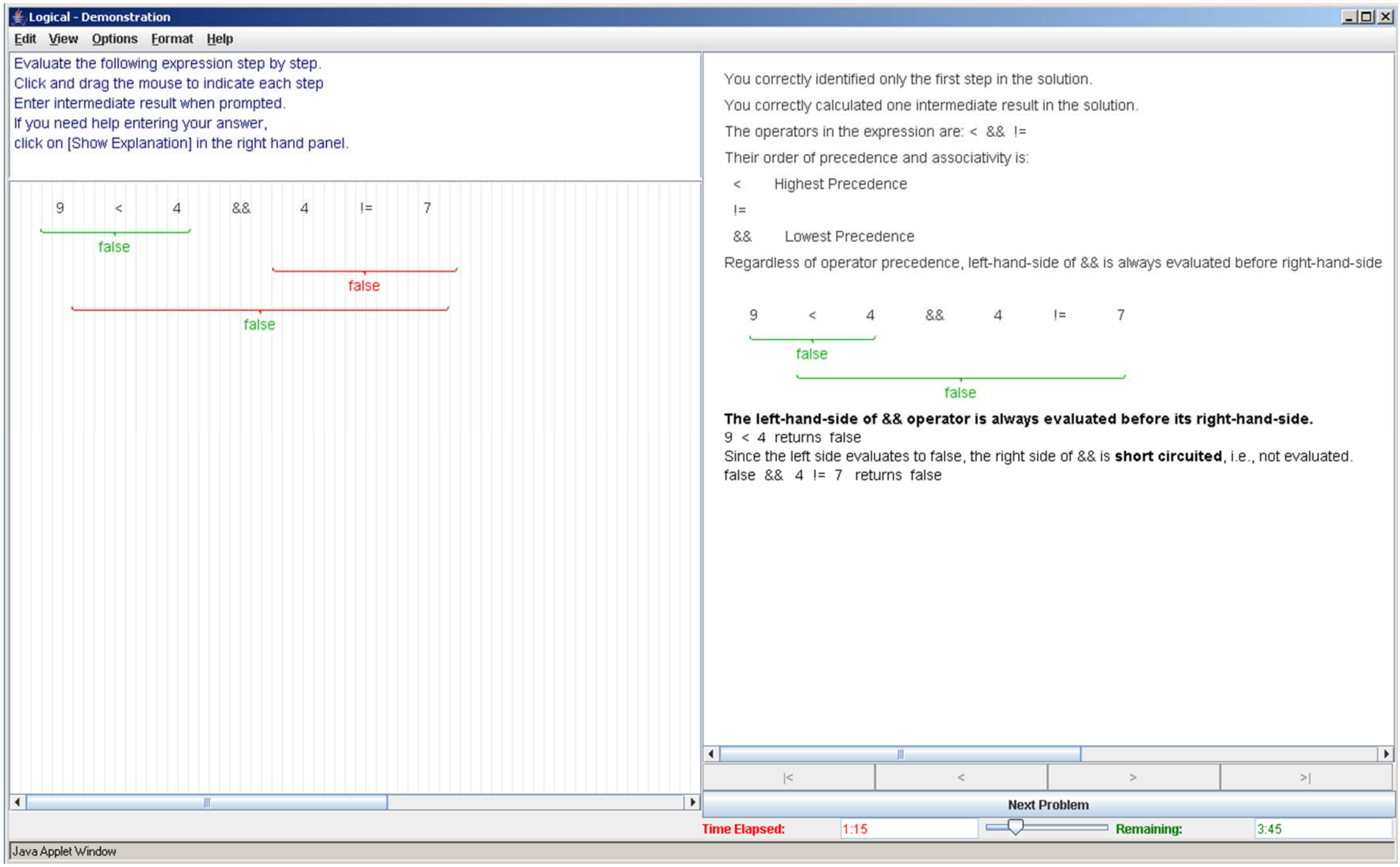
If interested, please contact:

amruth@ramapo.edu

Acknowledgements:

- NSF CCLI DUE 0088864

Problet on Expression Evaluation



Evaluate the following expression step by step.
Click and drag the mouse to indicate each step
Enter intermediate result when prompted.
If you need help entering your answer,
click on [Show Explanation] in the right hand panel.

You correctly identified only the first step in the solution.

You correctly calculated one intermediate result in the solution.

The operators in the expression are: < && !=

Their order of precedence and associativity is:

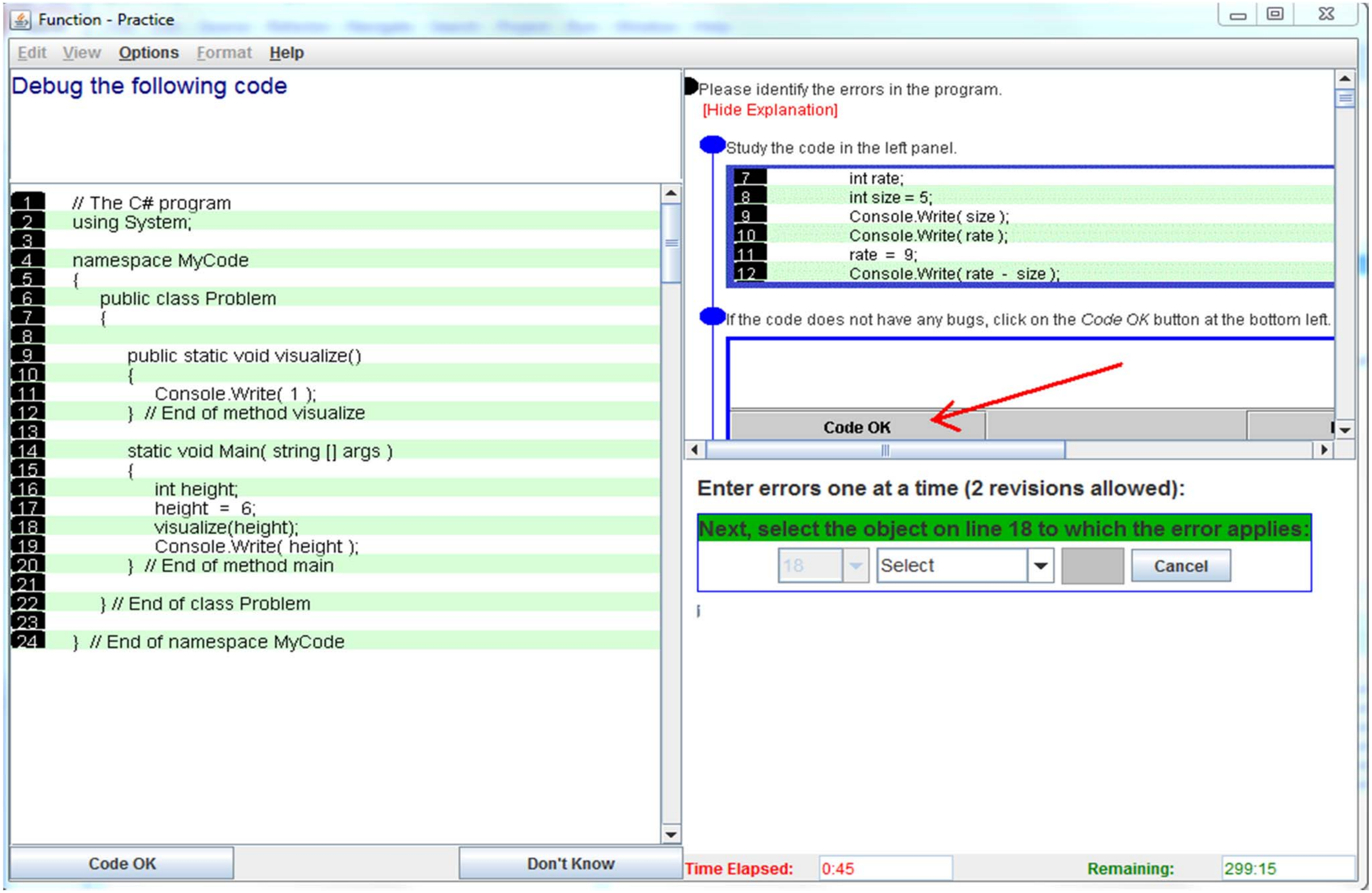
<   Highest Precedence
!=
&&   Lowest Precedence

Regardless of operator precedence, left-hand-side of && is always evaluated before right-hand-side

The left-hand-side of && operator is always evaluated before its right-hand-side.
9 < 4 returns false
Since the left side evaluates to false, the right side of && is **short circuited**, i.e., not evaluated.
false && 4 != 7 returns false

Time Elapsed: 1:15 Remaining: 3:45

Problet on Debugging



Debug the following code

```
1   // The C# program
2   using System;
3
4   namespace MyCode
5   {
6       public class Problem
7       {
8
9           public static void visualize()
10          {
11              Console.Write( 1 );
12          }  // End of method visualize
13
14          static void Main( string [] args )
15          {
16              int height;
17              height  =  6;
18              visualize(height);
19              Console.Write( height );
20          }  // End of method main
21
22      } // End of class Problem
23
24  }  // End of namespace MyCode
```

Please identify the errors in the program.

Study the code in the left panel.

```
7       int rate;
8       int size = 5;
9       Console.Write( size );
10      Console.Write( rate );
11      rate  =  9;
12      Console.Write( rate  -  size );
```

If the code does not have any bugs, click on the *Code OK* button at the bottom left.

Enter errors one at a time (2 revisions allowed):

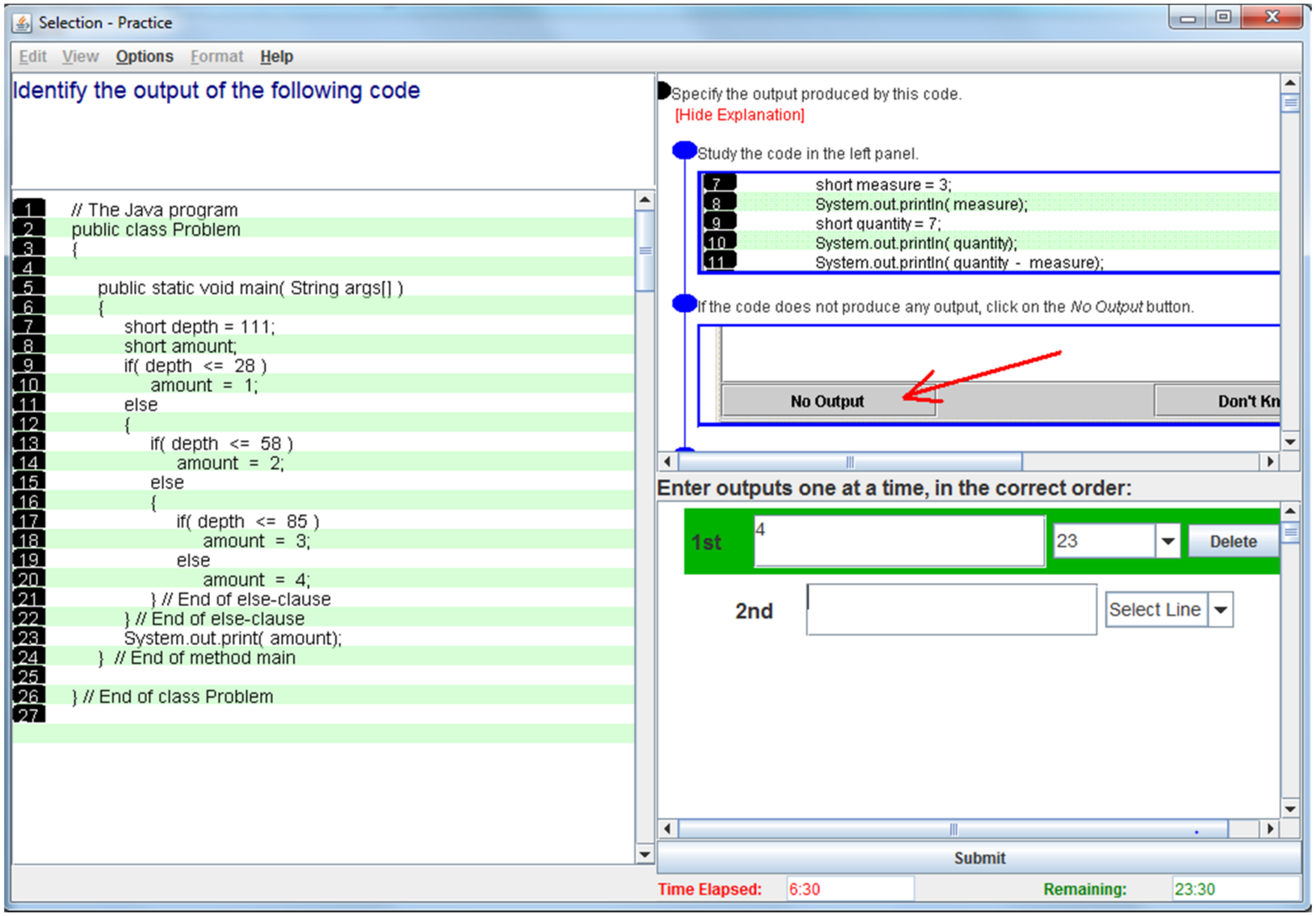
Next, select the object on line 18 to which the error applies:

18 | Select

Time Elapsed: 0:45

Remaining: 299:15

Problet on Predicting the Output



Identify the output of the following code

```
// The Java program
public class Problem
{

    public static void main( String args[] )
    {
        short depth = 111;
        short amount;
        if( depth  <=  28 )
            amount  =  1;
        else
        {
            if( depth  <=  58 )
                amount  =  2;
            else
            {
                if( depth  <=  85 )
                    amount  =  3;
                else
                    amount  =  4;
            } // End of else-clause
        } // End of else-clause
        System.out.print( amount);
    }  // End of method main

} // End of class Problem
```

Specify the output produced by this code.

[Hide Explanation]

Study the code in the left panel.

```
short measure = 3;
System.out.println( measure);
short quantity = 7;
System.out.println( quantity);
System.out.println( quantity  -  measure);
```

If the code does not produce any output, click on the *No Output* button.

Enter outputs one at a time, in the correct order:

1st 4 23

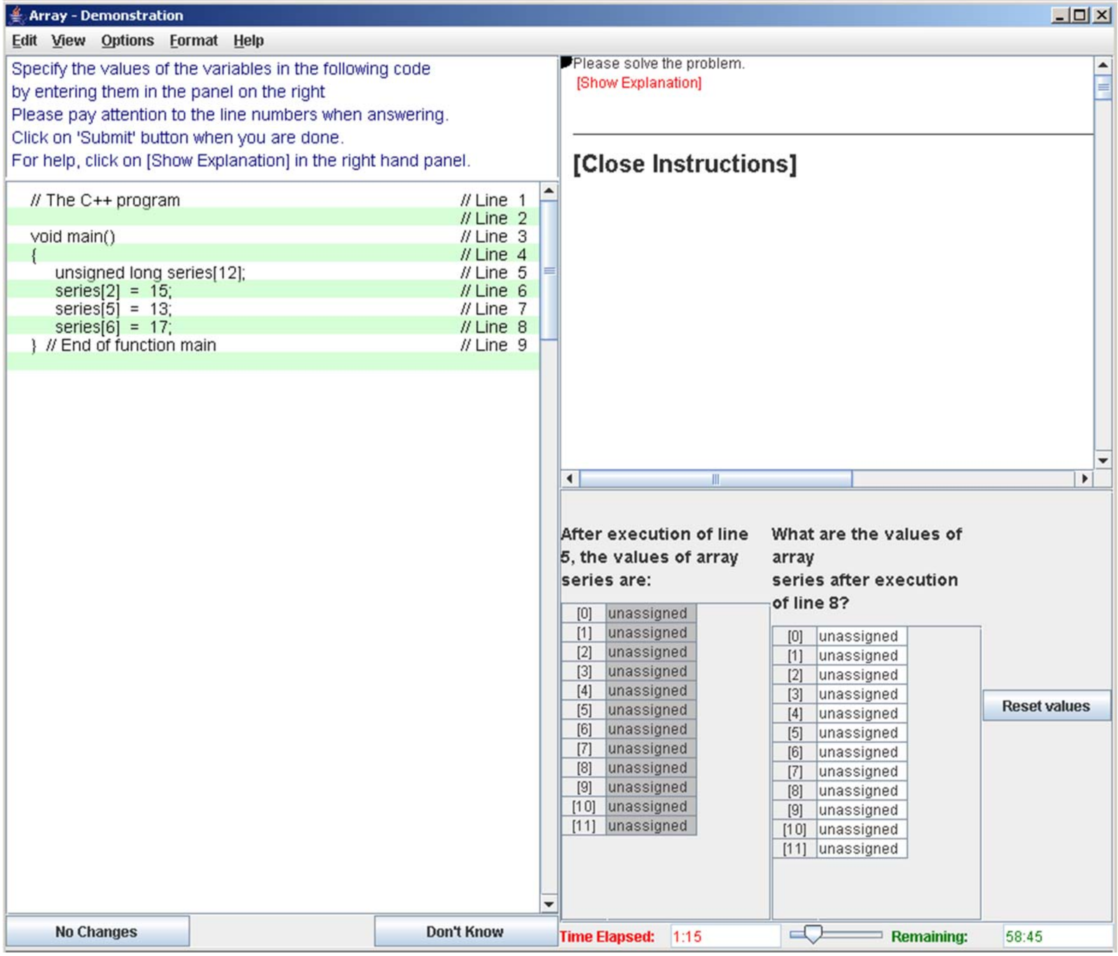
2nd Select Line

Time Elapsed: 6:30

Remaining: 23:30

Amruth Kumar, amruth@ramapo.edu

Problet on the State of a Variable



Specify the values of the variables in the following code by entering them in the panel on the right
Please pay attention to the line numbers when answering.
Click on 'Submit' button when you are done.
For help, click on [Show Explanation] in the right hand panel.

```
// The C++ program                         // Line  1
                                           // Line  2
void main()                                // Line  3
{                                          // Line  4
    unsigned long series[12];              // Line  5
    series[2]  =  15;                      // Line  6
    series[5]  =  13;                      // Line  7
    series[6]  =  17;                      // Line  8
}  // End of function main                 // Line  9
```

Please solve the problem.
[Show Explanation]

[Close Instructions]

After execution of line 5, the values of array series are:

[0]	unassigned
[1]	unassigned
[2]	unassigned
[3]	unassigned
[4]	unassigned
[5]	unassigned
[6]	unassigned
[7]	unassigned
[8]	unassigned
[9]	unassigned
[10]	unassigned
[11]	unassigned

What are the values of array series after execution of line 8?

[0]	unassigned
[1]	unassigned
[2]	unassigned
[3]	unassigned
[4]	unassigned
[5]	unassigned
[6]	unassigned
[7]	unassigned
[8]	unassigned
[9]	unassigned
[10]	unassigned
[11]	unassigned

Time Elapsed: 1:15
Remaining: 58:45