CEMC Math Circles - Grade 11/12

April 7, 2020 - April 13, 2020

Silly Square Roots

The connections between mathematics and computer science run deep. Here we explore these connections through the following problem which was Question 5a on the 2019 Euclid Contest. No programming experience is needed to follow along but there is also a challenge for you if you do have some programming experience.

Problem 1

Determine the two pairs of positive integers (a, b) with $a < b$ that satisfy the equation $\sqrt{a}+\sqrt{b} = \sqrt{50}$.

Solution to Problem 1 Using Mathematics

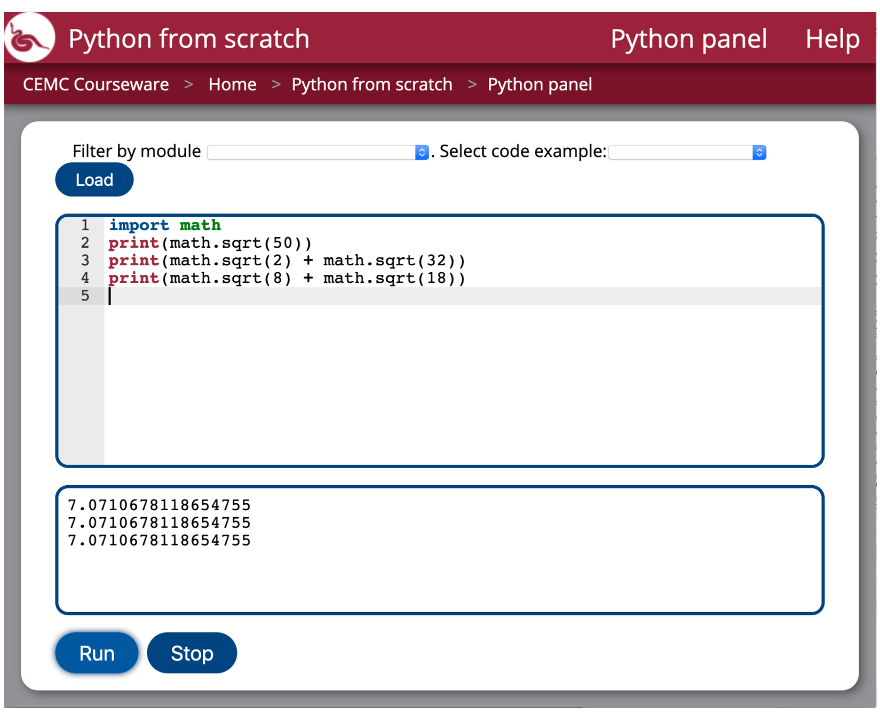
On a math contest or test, we would normally use known mathematical facts, such as the properties of square roots, to find the two pairs. Attempt the problem above this way and then check your answer. Note that your solution can rely on the fact that we were told there are exactly two pairs satisfying the equation.

Solution to Problem 1 Using Computer Programming

Now visit our Python from scratch panel and enter the following code in the upper code box.

```
import math
print(math.sqrt(50))
print(math.sqrt(2) + math.sqrt(32))
print(math.sqrt(8) + math.sqrt(18))
```

Hit the Run button to see the result of this program. Your screen should look something like this:



Congratulations if you just entered a computer program and ran it for the first time!

The key thing to observe is that this program uses a library function called `math.sqrt` to display the three values $\sqrt{50}$, $\sqrt{2}+\sqrt{32}$ and $\sqrt{8}+\sqrt{18}$. They all appear to be equal suggesting that the two pairs are $(2, 32)$ and $(8, 18)$. But where did these numbers come from?

In order to discover these two pairs, we could generate possible pairs (a, b) and check if $\sqrt{a}+\sqrt{b} = \sqrt{50}$ for each pair. Think about how you would generate possible pairs systematically and check whether the equality holds. The next program will simulate one way to do this. Replace your previous program with the following program and run it to see what it displays.

```
import math
for a in range(1, 50):
    for b in range(a + 1, 50):
        if (math.sqrt(a) + math.sqrt(b)) == math.sqrt(50):
            print((a,b))
```

Try to get a rough understanding of why this produces the right answer. One advantage of programs written in Python is that they tend to be readable by beginners. However, if you are new to programming or new to the language Python, here are a few notes to help explain some of the details:

- The first line gives the rest of the program access to a library of mathematical functions such as `math.sqrt`. If we do not include this line, Python will produce an error message.
- The second line tells us that a *variable* `a` will take on the integer values starting at 1 and ending at $50 - 1 = 49$. (Python does not include the second number in the range.) The block of three lines indented below this will be executed once for each of these values of `a`. Together, this is typically called a *loop*.
- The third line is similar to the second line except the values of `b` begin at the value `a + 1`. The result is that we have one loop nested inside another. For each value of `a`, the variable `b` will take on its own range of values.
- To see the how the values of the variables `a` and `b` change, you can `print` (or display) them at a strategic point in your program. Here is an example using 5 in place of 50:

```
for a in range(1, 5):
    for b in range(a + 1, 5):
        print((a,b))
```

```
(1, 2)
(1, 3)
(1, 4)
(2, 3)
(2, 4)
(3, 4)
```

- The fourth line executes a conditional test involving variables `a` and `b`. The `==` operator attempts to determine if the two expressions are equal. If they are, the indented line below is executed. Otherwise, it is skipped.

Now ask yourself why 50 is used as an upper bound in `range(1,50)` and `range(a + 1, 50)`. Can this be improved? Why is 1 used as a lower bound in `range(1, 50)` but not in `range(a + 1, 50)`? We will discuss these details when solutions to the next two problems are presented.

Problem 2

Determine the two pairs of positive integers (a, b) with $a < b$ that satisfy the equation $\sqrt{a}+\sqrt{b} = \sqrt{75}$.

Attempt to Solve Problem 2

The two pairs that solve this problem are $(3, 48)$ and $(12, 27)$. However, our programming solution to Problem 1 where each 50 is replaced by 75, does not work! You should try it yourself to see the result. Why is nothing displayed? We get a clue by doing this:

```python
import math
print(math.sqrt(75))
print(math.sqrt(3) + math.sqrt(48))
print(math.sqrt(12) + math.sqrt(27))

```

```
8.660254037844387
8.660254037844386
8.660254037844386
```

The fundamental problem is that the `math.sqrt` function does not compute exact values. We can see that the displayed results above are extremely close to each other but the first one, the purported value of $\sqrt{75}$, is just a bit larger.

This is not a flaw with Python! It is impossible to store let alone calculate irrational numbers like $\sqrt{75}$, π and e exactly. An infinite amount of memory would be needed to do this. We already know this from mathematics where we understand that an irrational number is one for which its decimal expansion does not terminate or end with a repeating sequence.

Can you find a way to use a combination of mathematical insight and Python to solve this second problem?

Problem 3

Determine *all* pairs of positive integers (a, b) with $a < b$ that satisfy the equation $\sqrt{a} + \sqrt{b} = \sqrt{147}$.

Looking for more?

See Piazza if you would like more challenges that combine mathematics and computer science.