LARSON—MATH 255–CLASSROOM WORKSHEET 04
Solving, Plotting, Definitions, Strings.

1. Create a Cocalc/Sage Cloud account.

   (a) Start the Chrome browser.

   (b) Go to `http://cocalc.com`

   (c) "Create new account" using **your VCU email address** .

   (d) You should see an existing Project for our class. Click on that.

   (e) Click "New", then "Sage Worksheet", then call it **c04**.

   (f) For each problem number, label it in theSAGE cell where the work is. So for Problem 1, the first line of the cell should be `#Problem 1`.

   Review

   - `solve` is SAGE's powerful and flexible command for solving systems of one or more equations.

2. Solve $x^2 + x = 25$.

3. Find all solutions of $\sin\theta = \frac{1}{2}$ by hand. Now evaluate `solve(sin(x)-0.5,x)`. Explain SAGE's result.

4. Define variables a, b and c. One way to do this is with the command `var("a b c")`. Solve $ax^2 + bx + c = 0$ by evaluating `solve(a*x**2+b*x+c, x)`.

5. Consider the following system. Sketch the graphs of these equations on the same coordinate system (by hand and then with `plot`), then `solve` to get the exact points of intersection. $\begin{cases} y = x^2 \\ y = x \end{cases}$

   More graphing and calculating basics

6. Make a point at $(4, 4)$ Evaluate `point((4,4))`.

7. Make it bigger by adjusting the "size" parameter; evaluate `point((4,4),size=200)`. Try other values for `size`.

8. Draw a line from $(-1, 1)$ to $(4, 4)$ by evaluating `line([(-1,1),(4,4)])`. Try drawing a line with 3 points.

9. Make the line thicker by adjusting the "thickness" parameter: evaluate `line([(-1,1),(4,4)],thickness = 4)`. Try other values of `thickness`.

10. Make the line dashed by adjusting the "linestyle" parameter: evaluate `line([(-1,1),(4,4)],linestyle="dotted")`. Try another value for "linestyle" by reading the options from the help command `line2d?`.

11. Now make the line red.

12. Draw a triangle between $(1,1)$, $(1,2)$, and $(2,1)$ using the line command.

13. Now draw a triangle between $(1,1)$, $(1,2)$, and $(2,1)$ using the `polygon` command; find examples of how this command works with `help(polygon)`. What's the difference?

14. Type in the following program and evaluate. (Note that there are *exactly* four spaces before the word "print").

```
def write_string(string_name):
    print(string_name)
```

    Now type `write_string("hello world!")` and evaluate.

    In order to do sophisticated calculations, or to allow for multiple inputs, you will need to write *programs*. Our "hello world!" program was the first example. It included a `print` statement. Other program features, in almost any language, include *conditional statements* (if..then..) and *loops*.

15. Type in the following function definition and evaluate.

```
def absolute(x):
    if x>=0:
        return x
    else:
        return -x
```

16. Now test it. Try `absolute(4)`, `absolute(-4)`, etc.

17. Now *use* the program you just wrote in another program. Evaluate and test the following.

```
def abs_plus_five(x):
    return absolute(x)+5
```

18. You don't have to add five, you can add *any* number by adding a *parameter*.

```
def abs_plus(x,y):
    return absolute(x)+y
```

19. Now test it. Try `abs_plus(4,5)`, `abs_plus(-4,5)`, `abs_plus(-4,23)`, etc.

20. Write your own function `triple_product` that takes *three* inputs (call them anything, or x, y, z is fine) and *returns* their product.

String formatting.

A *string* is a sequence of *characters* (letters, numerals, symbols, etc). If you put a sequence of characters between quotes, you are telling Sage to treat what's between the quotes as a string (instead of as a *keyword*). Strings can be manipulated, and have places that can be filled in.

21. Type and evaluate `print('This string has {}'.format('17 characters'))`. Now try replacing '17 characters' with any other string.

22. Type and evaluate the following program.

```
def superstring(x):
    print('This string has {}'.format(x))
```

23. Now test your function. Type and evaluate `superstring('black letters')`.

Getting your classwork recorded

When you are done, before you leave class...

(a) Click the "Make pdf" (Adobe symbol) icon and make a pdf of this worksheet. (If CoCalc hangs, click the printer icon, then "Open", then print or make a pdf using your browser).

(b) Send me an email with an informative header like "Math 255 - c04 worksheet attached" (so that it will be properly recorded).

(c) Remember to attach today's classroom worksheet!