Last name _____

First name _____

LARSON—MATH 305—CLASSROOM WORKSHEET 28
Sage LAB!—Primitive Roots

Organizational Notes

1. A Zoom recording link and class notes will be sent out after each Zoom class.
2. Remember to send your answers to the classroom worksheets, and pdfs of your Lab work. Title your email with enough to help me record your "participation".

Logging into Sage/CoCalc

1. Start the Chrome browser.
2. Go to `http://cocalc.com` and sign in.
3. You should see an existing Project for our class. Click on that.
4. Click "New", call it **c28**, then click "Sage Worksheet".
5. For each problem number, label it in the Sage cell where the work is. So for Problem 1, the first line of the cell should be `#Problem 1`.
6. When you are finished with the worksheet, click "make pdf", email me the pdf (at `clarson@vcu.edu`, with a header that says **Math 305 c28 worksheet attached**).

Primitive Roots

Consider $\mathbb{Z}/n\mathbb{Z}$. There are $\phi(n)$ units. We proved that each of these units has a *multiplicative order* (often just *order*). Recall the order of a unit a is the smallest exponent x such that $a^x \equiv 1 \pmod{n}$. We will prove that there is a unit a with order $\phi(n)$ (and explore the implications). Such a unit is called a *primitive root*.

1. Find a primitive root in $\mathbb{Z}/7\mathbb{Z}$ by hand. Now run: `primitive_root(7)`.

2. How could we write an algorithm to find primitive roots?

3. How many primitive roots does $\mathbb{Z}/p\mathbb{Z}$ (for prime p) have? Can you make a conjecture?

Fermat's Theorem & Primality Testing

We proved: if p is prime then $a^{p-1} \equiv 1 \pmod{p}$, for every $a \in [p-1]$. In fact, it can be shown that: p is prime if and only if $a^{p-1} \equiv 1 \pmod{p}$, for every $a \in [p-1]$.

4. Use this fact to write a primality test (that is a function which takes an integer n as input and outputs `True` if n is prime and `False` if it is not prime).

5. Try this function on some large composite numbers with two factors.

Polynomial Rings

6. We can use Sage to define the polynomial ring $R[x]$ corresponding to a ring R. Sage a little bit blurs R and $R[x]$: Defines a ring of polynomials named R over the ring of integers mod 12 by running: `R.<x>= PolynomialRing(R)`.

7. Run: `f=x^6-1`. We just defined x to be the x in a polynomial ring—so this polynomial is actually in R. Check.

8. Now let's check our theorem on the number of roots of this polynomail. Run: `f.roots()`.

Extras

9. If p and $p+2$ are both prime, they are called *twin primes*. Find the first several pairs of twin primes.