LARSON—MATH 353–CLASSROOM WORKSHEET 38
Conjecturing Invariant Bounds—Graph Theory

1. Log in to CoCalc.

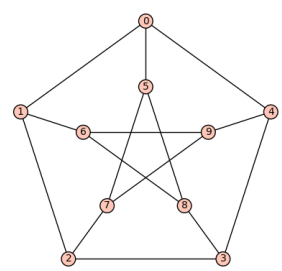
   (a) Start the Chrome browser.
   (b) Go to `https://cocalc.com`
   (c) Login (**your VCU email address** is probably your username).
   (d) You should see an existing Project for our class. Click on that.
   (e) Click "New", then "Worksheets", then call it **c38**.

We can use the CONJECTURING program to conjecture upper and lower bounds for an *invariant* of an mathematical object (number, matrix, graph, etc). An *invariant* in this context means any number associated with that object. So, for instance, the determinant of a matrix is a matrix-invariant.

Inequalities show up everywhere in mathematics; famous ones include the Cauchy-Schwartz inequality. Investigating bounds can be of enormous practical importance: bounds are useful when we want to reduce a *search space* where the answer to some question may be (for instance optimizing a discrete function).

Graph Theory

A *graph* consists of a set of *vertices* and *edges* between them. As an example consider the Petersen graph:



2. You can generate it in Sage with the command `pete = graphs.PetersenGraph()`. To view it run `pete.show()`

3. To get the vertices run: `pete.vertices()`. To get the edges run: `pete.edges(labels=False)`.

Graph Invariants

Graph invariants are numbers associated with graphs. The number of vertices is the *order*, while the number of edges is the *size*.

4. Run `pete.order()` and `pete.size()` to get the order and size of the Petersen graph.

Graph Classes

Graphs can have properties and the graphs with those properties form a class. A graph is *complete* is there is an edge between every pair of vertices.

5. Run `k5 = graphs.CompleteGraph(5)` for an example. View it, find its vertices, edges, size and order.

   A graph is *bipartite* if the vertices can be divided into two sets with no edges between any of the vertices in each of the sets.

6. Run `k_2_3 = graphs.CompleteBipartiteGraph(2,3)` for an example. View it, find its vertices, edges, size and order.

7. Run `k_5_3 = graphs.CompleteBipartiteGraph(5,3)` for an example. View it, find its vertices, edges, size and order.

   A graph is *triangle-free* if there is no triple of vertices with edges between each pair of the three vertices.

8. Check that $k_{2,3}$, $k_{5,3}$, and *pete* are all triangle-free.

9. All bipartite graphs are triangle-free. Why?

Extremal Graph Theory

A famous question: What is the largest number of edges a triangle-fee graph can have? Let's investigate by generating conjectures!

```
objects = [pete, k_2_3, k_5_3]

invariants = [Graph.size,Graph.order]

invariant_of_interest=invariants.index(Graph.size)

conjs=conjecture(objects,invariants,invariant_of_interest,upperBound=True)
for conj in conjs:
    print(conj)
```

10. **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 353—c38 worksheet attached" (so that it will be properly recorded).

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