Last name _____

First name _____

LARSON—OPER 731–SAGE WORKSHEET 07
Matroids!

1. Log in to your Sage Cloud account.

   (a) Start Chrome browser.

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

   (c) Click "Sign In".

   (d) Click project **Classroom Worksheets**.

   (e) Click "New", call it **s07**, then click "Sage Worksheet".

   Sage has built-in support for investigating matroids—all the types we've seen as well as any general matroid.

   Uniform Matroids.

2. First let's define a matroid with 5 elements in its ground set and independent sets with no more that 2 elements. Evaluate `M = matroids.Uniform(2, 5)`. Then evaluate M to see what Sage thinks M is.

3. Evaluate `M.groundset()` to get the groundset of M. (Sage uses *frozensets* instead of regular sets for technical reasons—they are stored in such a way that the sets can't be modified—and this allows for very fast access).

4. In general there are many more independent sets than bases (maximum rank independent sets). Instead of asking Sage to produce all independent sets we can get a picture of our matroid by asking for its bases. Evaluate: `M.bases()`.

5. Sage didn't actually do this. In general there are a *lot* of bases. Instead it produced an *iterator*—and object that produces bases one at a time. This is perfect for examining bases one at a time, and avoids the overhead of filling up the memory with all the bases. We can force Sage to give as all the bases by casting the iterator to a list. Evaluate: `list(M.bases())`.

6. If we want just a single base we can get that directly. Evaluate: `M.basis()`.

7. What is the rank of M? Evaluate: `M.rank()`.

8. First define a matrix A. Evaluate: `A=matrix(2,4,[1,0,0,1,2,3,4,5])`. Then evaluate: `A` to see this matrix.

9. We will define the linear matroid M whose independent sets are the sets of linearly independent columns of A. Sgae has a built-in matroid *constructor*: given many kinds of objects that you can define a matroid from, the constructor will produce the standardly-defined matroid. Evaluate: `M=Matroid(A)`. Then evaluate M to see what you have.

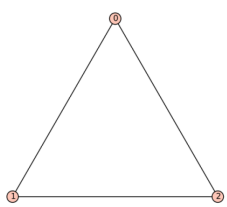
10. Evaluate: `M.groundset()`. What does the output mean?

11. Evaluate: `list(M.bases())`. What does the output mean?

12. Evaluate: `M.basis()`.

13. What is the rank of M? Evaluate: `M.rank()`.

Graphical Matroids.



14. The graph G above is the complete graph on 3 vertices. Evaluate `G = graphs.CompleteGraph(3)` to generate this graph. Evaluate: `G.show()` to see what you have.

15. The matroid constructor also works with graph inputs. Evaluate: `M = Matroid(G)`. Evaluate M to see what you have.

16. Evaluate: `M.groundset()`. What does the output mean?

17. Evaluate: `list(M.bases())`. What does the output mean?

18. Evaluate: `M.basis()`.

19. What is the rank of M? Evaluate: `M.rank()`.