Last name ______________________

First name ______________________

LARSON—OPER 731—BONUS WORKSHEET 13
Matroids in Sage.

1. Log in to your Sage Cloud account.

   (a) Start Chrome browser.

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

   (c) Click "Sign In".

   (d) Click the project for our course.

   (e) Click "New", call it **h13**, then click "Sage Worksheet".

   We will make the path graph P_3 with 3 vertices, the graphic matroid M corresponding to this graph, the linear matroid M_2 corresponding to the vertex-edge-incidence matrix of this graph (over GF(2), the field on 2 elements), and check that these are really the same matroid (isomorphic).

   Along the way we'll see other concepts we've talked about in class including the rank, bases, and the matroid polytope.

   ### P_3 example

2. The graph P_3 is built-in to Sage. We will construct it and give it the name $p3$. Run: `p3=graphs.PathGraph(3)`

3. Now let's view it to see what graph we have, how its vertices are labeled, etc. Run: `p3.show()`

4. Let's call it's vertex-edge incidence matrix A. Run: `A=p3.incidence_matrix()`. This just assigns A to the matrix. To see what you have, type A and run that. Check that it's the matrix you expect.

5. The matrix you have is, by default, a matrix over the reals. You can see that its not over GF(2) by running: `A+A`. What did you get?

6. Lets make A into a matrix over GF(2). Run: `A=matrix(GF(2),A)` Run: A to see what you have and then run: `A+A` to see that it's really over GF(2).

7. Now let's make the graphic matroid M. Run: `M=Matroid(p3)`.Evaluate M to see what you have.

8. What is the ground set and independent sets of M? Run: `M.groundset()` to get the groundset. What is in it?

9. Run: `M.independent_sets()` to get the independent sets of M. Is the output what you expect? Explain.

10. Run: `M.rank()` to get the rank of M. Is the output what you expect? Explain.

11. Run: `M.bases()` to get the bases of M. This returns an *iterator* (there are typically **lots** of bases and this keeps them from all being put into memory. We can force Sage to list the bases by running: `list(M.bases())` Is the output what you expect? Explain.

12. Now let's make the linear matroid M_2 from the columns of the vertex-edge incidence matrix A. Run: `M2=Matroid(A)`. Evaluate $M2$ to see what you have.

13. Find its groundset. Run: `M2.groundset()`. It doesn't list the columns—just the column numbers!

14. Find its independent sets. Run: `list(M2.independent_sets())`. Do these match up with the independent sets of the graphic matroid M? Explain.

15. Sage has a test to check if two matroids are really the same (isomorphic). To check if M and $M2$ are really the same, run: `M.is_isomorphic(M2)`.

Paw graph example

16. Let's make the paw graph, and repeat our construction of the corresponding graphic matroid and the linear matroid formed from the columns of its vertex-edge incidence matrix (interpreted over GF(2)). Run:

```
paw=graphs.CycleGraph(3)
paw.add_edge(0,3)
paw.show()
```

17. Construct the graphic matroid M corresponding to the paw graph.

18. Find the groundset of M.

19. Find the independent sets of M.

20. Find its vertex-edge incidence matrix A of the paw graph.

21. Force Sage to view A as a matrix over GF(2).

22. Construct the linear matroid $M2$ over A.

23. Find the groundset of $M2$.

24. Find the independent sets of $M2$.

25. Check that M and $M2$ are isomorphic.

Matroid Polytopes

26. We can make a polytope from a matroid: take the convex hull of the characteristic vectors of the independent sets. Run: `P=M.independence_matroid_polytope()`.

27. To find its vertices, run: `P.vertices()`