# CMSC 420: Coding Project 6
# Graph Partitioning

## 1 Due Date and Time

Due to Gradescope by - See Canvas etc. You can submit as many times as you wish before that.

## 2 Get Your Hands Dirty!

This document is intentionally brief and much of what is written here will be more clear once you start looking at the provided files and submitting.

## 3 Assignment

We have provided the template `cluster.py` which you will need to complete. More specifically you will fill in the code details to manage clustering tools associated to graphs. More details are given below.

# 4 Details

The class methods should do the following:

- `def addedge(self,edge):`

  Add the edge given by `edge` to the graph's Laplacian matrix.

- `def fiedlervector(self) -> np.array:`

  Calculate the Fiedler vector for the graph and return it.

- `def clustersign(self):`

  Calculate the two clusters for the graph using the Fiedler method. Return details are indicated in the template file.

# 5 Additional Functions

You probably don't need any additional functions.

# 6 What to Submit

You should only submit your completed `cluster.py` code to Gradescope for grading. We suggest that you begin by uploading it as-is (it will run!), before you make any changes, just to see how the autograder works and what the tests look like. Please submit this file as soon as possible.

# 7 Testing

This is tested via the construction and processing of tracefiles.

- The first line in the tracefile is `nodecount,n` which initializes an instance of the `Graph` class. It should set the `nodecount` equal to $n$ and initialize the `laplacian` matrix as an $n \times n$ matrix of zeros.
- Each remaining non-final line in a tracefile is `addedge,x,y` which indicates that an edge should be added between vertex `x` and vertex `y`.
- The final line is either `laplacianmatrix`, which prints the Laplacian matrix, `fiedlervector`, which prints the Fiedler vector, or `clustersign`, which clusters the graph.

Note that the testing suite first tests the Laplacian matrix so you should fix the `addedge` method first. It then tests the Fielder vector so you should fix the `fieldervector` method next. Finally it tests the clustering so that's when you should fix the `clustersign` method.

# 8 Local Testing

We have provided the testing file `test_cluster.py` which you can use to test your code locally. Simply put the lines from a tracefile (either from the autograder or just make one up) into a file `whatever` and then run:

```
python3 test_cluster.py -tf whatever
```