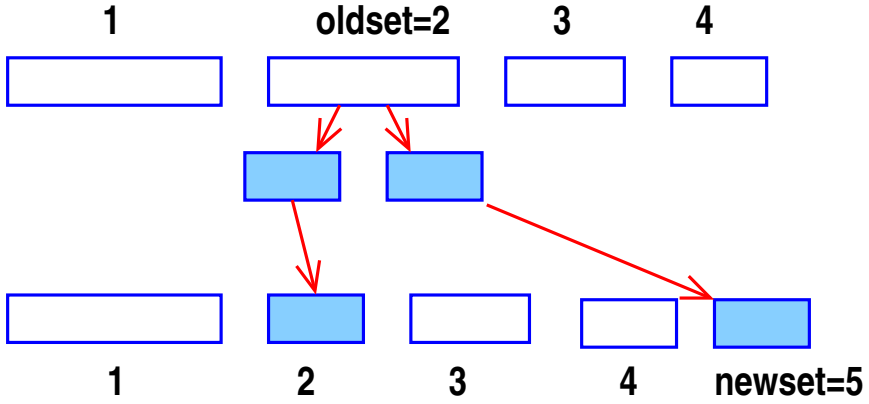
CSci 8314, S'19	Homework # 2	Due Date: April 17, 2019

1. In this homework you implement a basic recursive spectral bisection partitioning in matlab. Two-way bisection partitioning was demonstrated in class and the matlab diaries are available online from the class web-site. You will need to implement a recursive version of this technique which will keep partitioning the largest available partition at a given step, until a desired number of parts is reached. The interface to the function must be:

   ```
   function [Sets, labl] = RSB(A, maxsets)  .
   ```

   On input, `A` is a sparse matrix ~~graph Laplacean~~, and `maxsets` is the desired number of partitions. On return, `Sets` is a cell array with `Sets{i}` containing the list of nodes (indices) of partition i, and `labl` is an array of length n containing the labels (partition number) of the nodes. (The information provided by `labl` is redundant but it is returned because it will be available from the script `RSB`).

   The algorithm starts with a Laplacean A and a list of all the nodes in some array, e.g., `list = [1:n]`. The goal is to build an array of cells called `Sets` that will contain the partitions. Initially we have one set, so `Sets{1} = list`. At any given step we are partitioning a set, whose number is, say, `oldset` in two subsets. Initially, `oldset`=1. A matrix B is associated with the current list, starting with $B = A$. We compute the lowest 2 eigenvalues of B (using `eigs`), and get the Fiedler vector v. We partition `list` into two sets `listp` and `listm` according to the sign of $v - median(v)$. Thus, two sets are created. One of these sets is new, and is added to `Sets`: `Sets{newset}=listm`. The other one will just replace the current set that is being partitioned in two: `Sets{oldset} = listm`.



The process is illustrated in the above Figure where are already 4 partitions and the algorithm is in the process of partitioning set `oldset=2`. Note that at each step one partition is created and another one replaced.

You will be given a data set containing a finite elemenet mesh to partition. In the matlab web-page you will find scripts for reading the mesh and for plotting it (`mygplot`) and a couple of other utilities.

For this homework I will ask that you upload the following on Canvas:

(a) Your script `RSB`

(b) ... and any comments you may have [optional]

I will download your driver and RSB script and execute them - so make sure there is nothing missing. It is important to supply only ~~those 2 files~~ your script RSB.m mentioned above and to not make any changes to the other ones provided to you.