This course is an introduction to sparse matrix techniques with an emphasis on solving sparse linear systems of equations and eigenvalue problems. It will cover general methods that exploit sparsity, direct and iterative methods for solving sparse linear systems, sparse eigenvalue problems, and applications related to sparse matrix techniques. One of the goals of the course is to consider sparsity in contexts other than the classical one of the solution of partial differential equations. This will be a hands-on course. Students will learn about the algorithms and their complexity or convergence theory, and they will also get an understanding on how to implement them and how to work with sparse matrices in general. One of the requirements of the course is the completion of a term project. In addition, each student is also expected to give one lecture on a topic selected from a list.

A big part (2/3?) of the course will consist of traditional lectures given by the instructor [+ in-class exercises and 4 tests.] Another part will consist of the lectures covered by students taking the class. The course qualifies for a 50% project toward a plan C masters. If you intend to take this course to (partially) satisfy the project requirement of the plan C masters then I need to know about it. See section ‘Project + Presentation’ below for additional details and requirements.

- **Class Schedule:** Tu-Th 11:15 - 12:30 – Room: Bruininks Hall 123
- **Instructor:** Yousef Saad [http://www.cs.umn.edu/~saad][e-mail: saad@cs.umn.edu] Office: 5 -225B Keller H.; Ph: 612 -624 – 7804.
- **Prerequisite:** csci 5304 (or equivalent).
- **Office hours:** Mo Tu 1:30 pm – 2:30 pm, room 5-225B

Textbook

Part of the lectures will be based on my book: "Iterative methods for sparse linear systems (2nd edition)" which is posted (free download) – see [http://www.cs.umn.edu/~saad/books](http://www.cs.umn.edu/~saad/books)
This will be supplemented by articles and lecture notes. Supplemental reading: *Direct methods for sparse linear systems*, by T. A. Davis, SIAM publishing, 2006. (see also accompanying software CSparse noted in the resources link in the class web-site (see below). *You need not purchase the books listed above*. My own book is posted for free and will be only used for part of the material.

Note: Matlab/Octave will be often used for writing short programs. See the matlab section of the class web-page for information and some documentation.

**Lecture Notes**

In addition to the text, lecture notes on some of the chapters will be posted regularly. These will not be posted on moodle but in the cse classes web-site: [http://www-users.cselabs.umn.edu/classes/Spring-2017/csci8314/](http://www-users.cselabs.umn.edu/classes/Spring-2017/csci8314/) Click on the "Lect. Notes" icon in the menu. These may include material not available in the text-book. The lecture notes will be posted by topic rather than lecture by lecture. I try to have these posted in advance.

**Project + Presentation**

To pass this course, each student will need to complete a project and to give a short presentation. The presentation is to be selected from a list of about 30 articles that are listed under *Presentations*. These can be regarded as a part of the course material - dealing with topics of recent interest, specific applications, or issues related to efficient or parallel implementations.

You will also need to write a term paper. This will generally be on the use of sparse matrix techniques in specific application areas and it may or may not be related to the topic of the presentation you selected to give. You can for example expand on what you present and write a comprehensive survey on its topic. But you can also write a paper on how to exploit a specific class of sparse matrix methods in your own research. In the past the projects ranged from exhaustive survey articles on a specific topic (e.g., ‘sparse matrix techniques in text mining’) to implementations /comparisons of (a) specific method (s) say for solving sparse linear systems; or a specific theme in your own research that illustrates the use of sparse matrix methods. A few sample topics along with additional details, including grading criteria, what and how to present, and so on, will be given
toward the middle of the semester. Minimal requirements, as well as the details on grading criteria, will be specified later.

**Evaluation**

The evaluation of your performance in this class will be based on the following:

- Three tests at 10% each: 30% [actually three best out of four given tests.]
- Project: 30%
- Presentation: 30%
- Class participation and attendance: 10%

Note that the tests will be short tests that may resemble quizzes (duration ≈ 30mn) but they are programmed in advance and will have a very focused coverage. Final grades will be assigned based on the following scale, where T is the total score (out of 100) you achieved in the class.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Score Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100 ≥ T ≥ 92</td>
</tr>
<tr>
<td>A-</td>
<td>92 &gt; T ≥ 87</td>
</tr>
<tr>
<td>B+</td>
<td>87 &gt; T ≥ 83</td>
</tr>
<tr>
<td>B</td>
<td>82 &gt; T ≥ 77</td>
</tr>
<tr>
<td>B-</td>
<td>77 &gt; T ≥ 72</td>
</tr>
<tr>
<td>C+</td>
<td>72 &gt; T ≥ 65</td>
</tr>
<tr>
<td>C</td>
<td>65 &gt; T ≥ 60</td>
</tr>
<tr>
<td>C-</td>
<td>60 &gt; T ≥ 55</td>
</tr>
<tr>
<td>D+</td>
<td>55 &gt; T ≥ 50</td>
</tr>
<tr>
<td>D</td>
<td>50 &gt; T ≥ 40</td>
</tr>
<tr>
<td>F</td>
<td>40 &gt; T</td>
</tr>
</tbody>
</table>

For example, to get a B you will need a grade between 77 (inclusive) and 82 (exclusive).

**Overview of topics to be covered (tentative)**


7. Least-squares problems. QR (review), rank-revealing QR; Sparse QR.


9. The GMRES variants: GMRES, FGMRES, ORTHOMIN, ... Nonsymmetric Lanczos and related methods. The two-sided Lanczos algorithm. Bi-Conjugate gradient. CGS and bi CGSTAB.

10. Preconditioning. Preconditioned iterations: left, right, split preconditioning, PCG. SSOR preconditioner. ILU(0) and ILU(k). ILUT. Block preconditioners. Software: SPARSKIT, ITSOL, ILUPACK.


In addition, there will be additional topics selected from a list and covered by students. The list will be established later and will cover for example, text data, Latent Semantic Indexing and information retrieval; sparsity in optimization; non-negative matrix factorizations (NMF); Matrix completion; Overview of Domain Decomposition and/or Multigrid Methods; Graph partitioning; Hypergraphs; network graphs; ....