Aim

- Some of the most exciting work being done in the Algorithms and Complexity area today is concerned with the development of software which applies the algorithms and techniques to practical problems. Much progress has been made recently in graph drawing, computational geometry, timetable construction, etc.
- Real-life instances of these kinds of problems are typically too large to be solved without using efficient algorithms that have been developed for them.
- In this unit of study you will work in a group to develop a software product of this kind.

Groups

1. Form a group.
   - 4-5 students per group
2. Choose a name for your group
3. Choose a group leader for your group
4. Let me know about your group
   - Before August 2: Send one email to me
     - Name of the group
     - Name, SID and email address for each group member
   - On August 6: Hand in a hard copy of the above email message, signed by each group member.

- If you can’t form a group, send me an email
  - I will form a group for you
  - This should be done before August 2.

Group meetings

- Each group should meet at least once per week.
- Group leader should send minutes of the meeting by an email immediately after the meeting to me, with
  - A subject header “GG Minutes XX/YY/ZZ”, where GG is the group name and XX/YY/ZZ is the date of the meeting
  - A list of the people attending the meeting
  - One or two paragraphs describing the business of the meeting

Typical minutes of a meeting

- Subject: Golly Minutes 13/08/02
- To: comp3201@cs.usyd.edu.au (Comp3201)
- Date: Tue, 13 Aug 2002 12:42:47 +1000 (EST)
- From: freddy@ugrad.cs.usyd.edu.au
- Cc: harry@staff.cs.usyd.edu.au, lollie@ugrad.cs.usyd.edu.au, mollie@ugrad.cs.usyd.edu.au, ollie@ugrad.cs.usyd.edu.au, collie@ugrad.cs.usyd.edu.au
- Attendees: Fred Nurk, Lollie Madsen, Ollie Owsome
- Business: We discussed the progress against the plan that was decided last week. Olle has produced a simple user interface, but it has no functionality at the moment. Collie is sick and didn’t attend. We decided that we should try to use OpenGL for the graphics.
Assessment

- Minutes: 10%
- Mid-term Report (35%): presentation and written report
  - presentation (including demo) and written report
  - graded by effectiveness & efficiency
- Marks
  - As a default, all students in the same group will be given the same mark (if some variation is appropriate, then the group should all make appointments to talk with me).
  - There may be a (shared) presentation for each group member if there is some disagreement within the group about the distribution of marks.
- Note: Plagiarism or any other form of dishonesty will be dealt with by standard University procedures.

Project description

- Symmetric Crossing minimization (SCM) problem
  - We want to construct a drawing of a graph which displays given symmetries with minimum number of edge crossings.
  - This is NP-hard problem.
  - In this project, we want to find good heuristics to solve this problem. Then we implement those heuristics and evaluate them by experiments.

Example

- Orbit: equivalence class of vertices
- Inter-orbit/ intra orbit edges
- Optimal drawing (1 crossings)
- Non-optimal drawing (5 crossings)

SCM project

- Find good heuristics for symmetric crossing minimization problem
  - You may need to modify the heuristics to apply
- Implement each heuristics
  - Each group will be implementing
    - Two heuristics in common
    - One their own heuristics
- Evaluate each heuristics
  - Test data will be given
  - Present experimental result: quality & time
- Visualisation: construct a drawing

Time Table

- Jul 30, Aug 6: lectures (by me)
- Investigate heuristics (4 weeks)
- Sep 10:
  - Each group present their heuristics
  - Discuss & select some heuristics
- Implementation (4 weeks)
- Oct 15: last lecture (by me)
  - How to write a report
  - Provide test data set
- Experiment + visualisation (3 weeks)
- Nov 5: Demo & presentation of the system