COMP 5900I - Fall 2026

Graphs and Algorithms

Course Information

The course is delivered online.

Activity	Day(s)	Time (Eastern Time)
Lectures	Tuesday and Thursday	10:00 am–11:30 am
Tutorial	Wednesday	1:00 pm–3:00 pm

Instructor	Office	Email
Prosenjit Bose	5302 Herzberg	`jit@scs.carleton.ca`
Bobby Miraftab	Online / by appointment	`bobby.miraftab@gmail.com`

Office hours: by appointment.
Best ways to be in touch: in class, by email, or during office hours.
Teaching assistants and their contact/office-hour information will be posted after the course begins.

Course website: https://www.bobby-miraftab.com/teaching/graphs-and-algorithms

The delivery of course material will consist of the following:

- **Lectures** are live online meetings.
- **Tutorial sessions** provide additional discussion, problem solving, and TA assistance.
- **Weekly assignments** contain approximately 4–5 problems emphasizing proofs, structural reasoning, and algorithmic analysis. Unless otherwise announced, assignments are due on **Fridays at 12:00 pm (noon), ET**.
- **Project and paper presentations** are described below.
- The **final examination** is an **online oral examination**. Scheduling and detailed instructions will be announced in advance.

Course Description

This course provides a graduate-level exploration of graph algorithms and structural graph theory. The emphasis is on algorithm design, mathematical analysis, and theoretical foundations of problems on graphs. Topics include graph traversal, spanning trees, network flows, matchings, graph coloring, planar graphs, and advanced topics such as geometric graphs, spectral graph theory, random graphs, and tree-decompositions.
The course blends proofs of correctness and complexity with structural insights from modern graph theory. No advanced programming experience is required; basic familiarity with Python and the `networkx` library may be useful.

Learning Outcomes

Upon successful completion of the course, students should be able to:

- formally analyze classical and modern graph algorithms;
- prove correctness and derive complexity bounds for graph algorithms;

- apply structural graph theory techniques to algorithmic problems;
- understand and use geometric and spectral methods on graphs; and
- read, present, and critically assess research papers in graph algorithms.

Prerequisites and Preclusions

This is a graduate special-topics course. Students are expected to have a working knowledge of discrete mathematics and basic algorithms.

Learning Materials and Reference Texts

There is no required textbook, and students are not required to purchase textbooks or other learning materials for this course. Useful references include:

- T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*.
- J. Kleinberg and E. Tardos, *Algorithm Design*.
- R. Diestel, *Graph Theory*, 6th edition, Springer.
- A. Brandstädt, V. B. Le, and J. P. Spinrad, *Graph Classes: A Survey*, SIAM, 1999.
- M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, 2nd edition, 2004.
- H. L. Bodlaender and A. M. C. A. Koster, *Combinatorial Optimization on Graphs of Bounded Treewidth*.
- Selected research papers and surveys assigned during the course.

Topics Covered

Week	Topic / Content
1	Graph definitions and representations: basic terminology, adjacency lists and matrices, connectivity, and motivating examples.
2	Graph traversal, connectivity, and spanning trees: BFS, DFS, connected components, cut vertices and bridges, strongly connected components, topological sorting, trees and forests, Kruskal and Prim.
3	Network flows and minimum cuts: Ford–Fulkerson, augmenting paths, max-flow min-cut, Dinic, Push–Relabel, and bipartite matching.
4	Advanced matching algorithms: Hopcroft–Karp, overview of Edmonds' Blossom algorithm, assignment problems, weighted matching, and stable matching.
5	Graph coloring and planar graphs: vertex coloring, greedy methods, complexity of coloring, Euler's formula, Kuratowski's theorem, embeddings, and duality.
6	Planar separators and Schnyder woods: separator theorems, balanced separators, Schnyder woods, and structural applications.
7	Geometric graphs: unit disk and intersection graphs, Delaunay triangulations, Voronoi diagrams, WSPD spanners, and applications.
8	Geometric and spectral graph theory: Theta and Yao graphs; adjacency and Laplacian matrices; eigenvalues, algebraic connectivity, and spectral clustering.

9	Random graphs: classical models, connectivity thresholds, and probabilistic methods.
10	Tree-decompositions and treewidth: bags, adhesion, width, treewidth, standard examples, and tree-likeness.
11	Structure of tree-decompositions: equivalent characterizations, separators, balanced separators, optimal decompositions, and nice tree-decompositions.
12	Algorithms on bounded-treewidth graphs: dynamic programming for Vertex Cover, Independent Set, Dominating Set, and k-Coloring; overview of algorithmic meta-theorems.

The schedule may be adjusted slightly to reflect the pace of the class and university holidays or breaks.

Evaluation

Final grades will be determined according to the following scheme:

Component	**Grade Value**
Weekly Problem Sets	50%
Project and Paper Presentation	25%
Online Oral Final Examination	25%
Total	**100%**

Project and Paper Presentation

- You can form a team of at most **2 members** for a project.
- **Written Project Proposal:** Due by **September 30th**. This is a brief submission used to choose a topic and reserve a paper.
- **End-of-Term Project Presentation:** Scheduled within the last **3–4 weeks of classes**.
- **Project Report:** Due a couple of days before your end-of-term project presentation.

Outline is as follows:

1. **Pick a topic.** The idea is to summarize the article in your own words, and possibly implement some aspect of it. A list of papers is available here Students may also propose a different paper or research topic, subject to instructor approval.
   (a) *Every Planar Graph Is the Intersection Graph of Segments in the Plane*, Chalopin and Gonçalves, 2009.
   (b) *On Grid Intersection Graphs*, Hartman, Newman, and Ziv, 1991.
   (c) *Triangle Contact Representations and Duality*, Gonçalves, Lévêque, and Pinlou, 2010.
   (d) *Triangle-Free Planar Graphs as Segment Intersection Graphs*, de Castro, Cobos, Dana, Márquez, and Noy, 2002.
   (e) *Computing the Independence Number of Intersection Graphs*, Fox and Pach, 2011.
   (f) *Unit Rectangle Visibility Graphs*, Dean, Ellis-Monaghan, Hamilton, and Pangborn, 2008.
   (g) *Graph Searching, and a Minimax Theorem for Tree-Width*, Seymour and Thomas, 1993.
   (h) *Schnyder Greedy Routing Algorithm*, He and Zhang, 2010.

(i) *Schnyder Embeddings for Non-Triangulated Graphs*, Felsner, 2004.
(j) *Map Graphs*, Chen, Grigni, and Papadimitriou, 2002.
(k) *New Linear-Time Algorithms for Edge-Coloring Planar Graphs*, Cole and Kowalik, 2008.
(l) *Contact Representations of Planar Graphs with Cubes*, Felsner and Francis, 2011.
(m) *Treewidth and Pathwidth of Permutation Graphs*, Bodlaender, Kloks, and Kratsch, 1995.
(n) *On Coloring Unit Disc Graphs*, Gräf, Stumpf, and Weißenfels, 1998.
(o) *Planar Graphs Have Bounded Queue-Number*, Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood, 2020.
(p) *A Bound on the Chromatic Number of the Square of a Planar Graph*, Molloy and Salavatipour, 2005.
(q) *Tree Decompositions Whose Trees Are Subgraphs: An Application of Simon's Factorization*, Bourneuf, Joret, Micek, Milanič, and Pilipczuk, 2026.
(r) *Reconfiguration in Bounded Bandwidth and Tree-Depth*, Wrochna, 2018.
(s) *Computing with Tangles*, Grohe and Schweitzer, 2015.
(t) *On Shortest Path, BFS- and DFS-Tree Graphs*, Bose, Madani, Maheshwari, and Miraftab, 2026.
(u) *On the Complexity of Secluded Path Problems*, Hanaka and Tsuru, 2025.

2. **Written Project Proposal.** By **September 30th**, submit the title of your chosen topic or paper and the names of the team members. The purpose of this short submission is to choose the topic and reserve the paper. Students proposing a paper or topic that is not on the list must obtain instructor approval.

3. **Final Project Presentation.** The presentation should be approximately **12–15 minutes** long. It will be held during the last **3–4 weeks of the course**.

4. **Project Report.** The report should have the format of a research article. It is best to use LaTeX (for example, Overleaf). The report should include:
   (a) **Introduction:** Motivation, problem statement, related work, and a short summary of what you did.
   (b) **Preliminaries:** Notation, definitions, and other necessary background.
   (c) **Main Section:** How you solved the problem; state the algorithm, its analysis, and its correctness.
   (d) **Experiments:** Include this section if you performed simulations or computational experiments.
   (e) **Conclusions:** Summary, what you learned, and what you think can be done in the future.
   (f) **References.**
   (g) The report should be approximately **4 pages long** and will be posted on the course web page.
   (h) You may use a double-column format. For example, you may use the Canadian Conference in Computational Geometry (CCCG) style file available here: http://vga.usask.ca/cccg2020/CCCG2020-tex-template.zip

Assignment Submission and Late Work

Unless otherwise announced, weekly assignments are due on **Fridays at 12:00 pm (noon), ET**. Late assignments and late project submissions lose **20% of the available mark for each day late**. A submission that is five or more days late therefore receives no credit unless an approved

academic consideration applies.
Students requiring short-term or long-term academic consideration should follow the applicable Carleton University procedures and contact the instructors as soon as reasonably possible.

Final Examination

The final examination is an **online oral examination** worth 25% of the final grade. The examination schedule, expected duration, and detailed procedure will be announced in advance. Students are expected to answer individually and to explain their reasoning clearly.

Use of AI and Machine-Learning Tools

Students may use **any machine-learning system and any AI model or tool**, including generative-AI systems, in completing assignments and project work. Students remain responsible for the correctness, clarity, and integrity of everything they submit. Any external sources, including AI-generated material when substantively used, should be acknowledged in a manner appropriate to the work.
The online oral final examination is an individual assessment; any specific rules governing tools or external assistance during the examination will be stated in the examination instructions.

Collaboration and Academic Integrity

Discussion of course material is encouraged. Unless an assessment explicitly permits group work, submitted solutions must represent the student's own understanding. Project work may be completed by teams of at most two students.
Students must follow Carleton University's Academic Integrity Policy. Sharing or posting course assessments or other restricted course materials online without permission is prohibited. If you are unsure whether a particular form of collaboration, citation, or tool use is permitted, ask an instructor before submitting the work.

Graduate Student Resources

University of Ottawa graduate students requiring Brightspace access should consult the Carleton graduate-student access information. Students requiring SCS IT resources should contact SCS Technical Support using their Carleton email account and identify the required resource and course. The School of Computer Science graduate advisors are available in Room 5302 Herzberg and at `grad.scs@carleton.ca`.

Academic Accommodations

Carleton University is committed to academic accessibility. Students who require academic accommodations or considerations should follow the procedures described at:
https://students.carleton.ca/course-outline/

Student Concerns

If you have a concern regarding this course, your first point of contact should be the instructors. If the matter cannot be resolved, the next point of contact is the School of Computer Science, followed by the Office of the Dean of Science, in accordance with University procedures.