THE UNIVERSITY OF WINNIPEG

APPLIED COMPUTER SCIENCE

Course Number: GACS-7101-002
Course Name: Advance Data Structures and Algorithms
Course Webpage: http://courses.acs.uwinnipeg.ca/7101/

Instructor Information

Instructor: Dr. Mary Adedayo
E-mail: m.adedayo@uwinnipeg.ca

Office Hours:	Tues/Thurs	11:30 – 12:30 pm	**Office:**	3D27
Class Meeting Times:	Tues/Thurs	10:00-11:15 am	**Room:**	3D03

Important Dates

1.	First Class:	Tuesday, September 3, 2019
2.	Reading Week (no classes):	October 13-19, 2019
3.	Midterm Test:	**Tuesday, October 22, 2019**
4.	Project announcement[2]:	Tuesday, November 5, 2019
5.	Final Withdrawal Date[1]:	Tuesday, November 12, 2019
6.	Last Class:	Thursday, November 28, 2019
7.	Final Exam[2]:	(See note)
8.	Project topics announcement:	**Tuesday, November 5, 2019**
9.	Deadline for Project reports:	**Thursday, December 5, 2019 at 11:59pm**
10.	Other Deadlines[3] (Assignments):	(See note)
11.	University closures: Thanksgiving	Monday, October 14, 2019
	Remembrance Day	Monday, November 11, 2019

[1]Final withdrawal date without academic penalty: A minimum of 20% of the work on which the final grade is based will be evaluated and available to the student before the voluntary withdrawal date.
[2]The final examination will be replaced with a project, for which the students are required to implement some challenging algorithms (using either C++ or JAVA) and make tests and comparisons. Details is provided in the Evaluation criteria and Guide to the Project section of this outline.
[3]Deadlines for assignments will be communicated during the course.

Course Objectives / Learning Outcomes

In this course, students will study methods for designing efficient data structures and algorithms such as binary search trees, red-black trees, priority queues, minimum spanning trees, strongly connected components, maximum flows, string matching and tree matching, as well as bipartite graphs. Through the study of these data structures and algorithms, students will develop skills to solve hard problems in specialized databases such as Web and Document, DNA and Deductive Databases.

Evaluation Criteria

1. Assignments (25%)
   - 3 assignments (weighted 7%) and 1 presentation (weighted 4%).
   - Due date will be specified on each assignment.
   - Assignments should be completed in English, typed and handed in through e-mail to the instructor before the deadline.
   - Assignments will be accepted up to 24 hours late with a 25% penalty.
   - Each student will give a 20-minutes presentation during 1 of the lectures. The content and the date of each presentation will be announced during the lectures. Each student will be assigned to a particular presentation date. The content and student assigned to each presentation will be announced during the lectures.

2. Midterm Test (20%)
   - During the regular class time on Tuesday, October 22, 2019.

3. Final Exam (55%)
   - The final exam will be replaced by a project, for which the students are required to implement some challenging algorithms (using a programming language), make tests and comparisons and write a report of between 15 and 25 pages.
   - The project will be evaluated by its originality and novelty (8/55), technical soundness and completeness of the implementation (10/55), adequacy of discussions and experiments about the implementation (10/55), ability to apply and implement the solution to a specific problem domain (10/55), readability and organization of the typed report (8/55), and Presentation of the final project (9/55).
   - The projects will be completed individually. Project topics will be announced on Tuesday, November 5, 2019.
   - The deadline for the project report submission will be on December 5, 2019 at 11:59pm.
   - Project report will be accepted up to 24 hours late with a 25% penalty.
   - The project will be presented in a 30-min presentation on December 13, 2019.
   - See the Guide to the Project section for more information about the project.

Students should contact the instructor as soon as possible if extenuating circumstances require missing an assignment, test or examination. A medical certificate from a practicing physician may be required before any adjustments are considered.

Test / Exam Requirements

- No Photo ID is required.
- The midterm test is closed book. No equipment (e.g. calculators, dictionaries, handheld devices, cell phones) are authorized for use in test.
- The textbook and any other equipment can be used for the final exam/project.

Final Letter Grade Assignment

Historically, numerical percentages have been converted to letter grades using the following scale. However, instructors can deviate from these values based on pedagogical nuances of a particular class, and final grades are subject to approval by the Department Review Committee.

A+	90 – 100%	B+	75 – 79%	C	60 – 64%
A	85 – 89 %	B	70 – 74%	D	50 – 59%
A-	80 – 84%	C+	65 – 69%	F	below 50%

Required Text Book / Reading List

- Cormen, Leiserson, Rivest and Stein, *Introduction to algorithms, 3rd Edition,* The MIT Press, 2009.

Besides the information contained in the main text, the instructor may also distribute papers, and discuss appropriate material and examples from other sources. Students are responsible for all material covered in the class.

Optional Text Book (Used for Reference only)

- Jeffrey D. Ullman, Alfred V. Aho, and John E. Hopcroft, *The Design and Analysis of Computer Algorithms,* Addison-Wesley Publishing Com., London, 1969

Prerequisite Information

- Consent of Department Graduate Program Committee Chair or Instructor.

Services for Students

Students with documented disabilities, temporary or chronic medical conditions, requiring academic accommodations for tests/exams (e.g., private space) or during lectures/laboratories (e.g., note-takers) are encouraged to contact Accessibility Services (AS) at 204-786-9771 or accessibilityservices@uwinnipeg.ca to discuss appropriate options. All information about a student's disability or medical condition remains confidential.
https://www.uwinnipeg.ca/accessibility-services.

Students may choose not to attend classes or write examinations on holy days of their religion, but they must notify their instructors at least two weeks in advance. Instructors will then provide

opportunity for students to make up work examinations without penalty. A list of religious holidays can be found in the 2019-20 Undergraduate Academic Calendar online at http://uwinnipeg.ca/academics/calendar/docs/important-notes.pdf

All students, faculty and staff have the right to participate, learn, and work in an environment that is free of harassment and discrimination. The UW Respectful Working and Learning Environment Policy may be found online at https://www.uwinnpeg.ca/respect.

Misuse of Computer Facilities, Plagiarism, and Cheating

Academic dishonesty is a very serious offense and will be dealt in accordance with the University's policies.

Avoiding Academic Misconduct and Non-academic Misconduct. Students are encouraged to familiarize themselves with the Academic Regulations and Policies found in the University Academic Calendar at:
https://uwinnipeg.ca/academics/calendar/docs/regulationsandpolicies.pdf
Particular attention should be given to subsections 8 (Student Discipline), 9 (Senate Appeals) and 10 (Grade Appeals). Please note, in particular, the subsection of Student Discipline pertaining to plagiarism and other forms of cheating.

Detailed information can be found at the following:

- Academic Misconduct Policy and Procedures: https://www.uwinnipeg.ca/institutional-analysis/docs/policies/academic-misconduct-policy.pdf and https://www.uwinnipeg.ca/institutional-analysis/docs/policies/academic-misconduct-procedures.pdf
- Non-Academic Misconduct Policy and Procedures: https://www.uwinnipeg.ca/institutional-analysis/docs/student-non-academic-misconduct-policy.pdf and https://www.uwinnipeg.ca/institutional-analysis/docs/student-non-academic-misconduct-procedures.pdf

Misuse of Filesharing Sites. Uploading essays and other assignments to essay vendor or trader sites (filesharing sites that are known providers of essays for use by others who submit them to instructors as their own work) involves "aiding and abetting" plagiarism. Students who do this can be charged with Academic Misconduct.

Avoiding Copyright Violation. Course materials are owned by the instructor who developed them. Examples of such materials are course outlines, assignment descriptions, lecture notes, test questions, and presentation slides. Students who upload these materials to filesharing sites, or in any other way share these materials with others outside the class without prior permission of the instructor/presenter, are in violation of copyright law and University policy. Students must also seek prior permission of the instructor /presenter before photographing or recording slides, presentations, lectures, and notes on the board.

Students are strongly recommended to view the University of Winnipeg library video tutorial Avoiding Plagiarism, which is available at: https://www.youtube.com/watch?v=UvFdxRU9a8g

Class Cancellation, Correspondence with Students and Withdrawing from Course

When it is necessary to cancel a class due to exceptional circumstances, the course instructor will make every effort to inform students via uwinnipeg email (and/or using the preferred form of communication, as designated in this outline), as well as the Departmental Assistant and Chair/Dean so that class cancellation forms can be posted outside classrooms.

Students are reminded that they have a responsibility to regularly check their uwinnipeg e-mail addresses to ensure timely receipt of correspondence from the University and/or the course instructor**.**

Please let course instructor know if you plan on withdrawing from the course. Note that withdrawing before the VW date does not necessarily result in a fee refund.

Topics to be covered (tentative)

1. Algorithm basics
2. Divide and conquer
3. Sorting algorithms
4. Basic data structures and Hash tables
5. Binary search tree and Red-black trees
6. Dynamic programming
7. Greedy algorithms
8. B-Trees
9. Graph algorithms
10. Elementary Graph algorithms and Minimum spanning trees
11. Single source shortest paths and All-pairs shortest paths
12. Maximum flow
13. String matching

Guide to the Project

The report should be your own work and should contain at least the following sections:

1. Introduction: this should include the problem description, motivation or significance of the problem and its application in computing or other real world domain.
2. Related work: this should describe the important techniques related to the problem to be addressed and discuss how your report fits in with such work.
3. Discussion: this can subdivided into different sections as you wish but should contain a detailed description of the method or work done, the formal algorithm or implemented, and analysis of the computational complexities, including time and space overhead.

4. Experiments: discuss some of the main data structures used for implementation, data used for the tests, and the test results using charts, histograms, tables or others. You must create a GitHub public repository for your implementation and include the link to it in your report. It is your responsibility to ensure the link is accessible.
5. Applications: discuss some of the applications of your solution or the implemented algorithm in the computing industry or to real world problems and how your code can be adapted to solve them.
6. Future work: a discussion of possible improvements, or possible extension of the problem or the solution provided.
7. References: ensure that related works and any source(s) of significant use in the project. Note that the information on plagiarism applies to every part of the project including the codes, report and experiments.