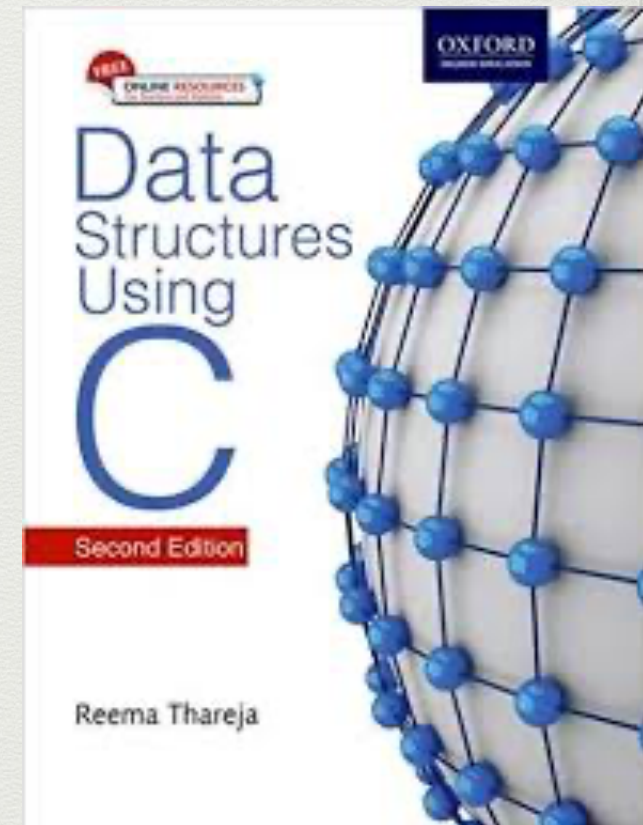
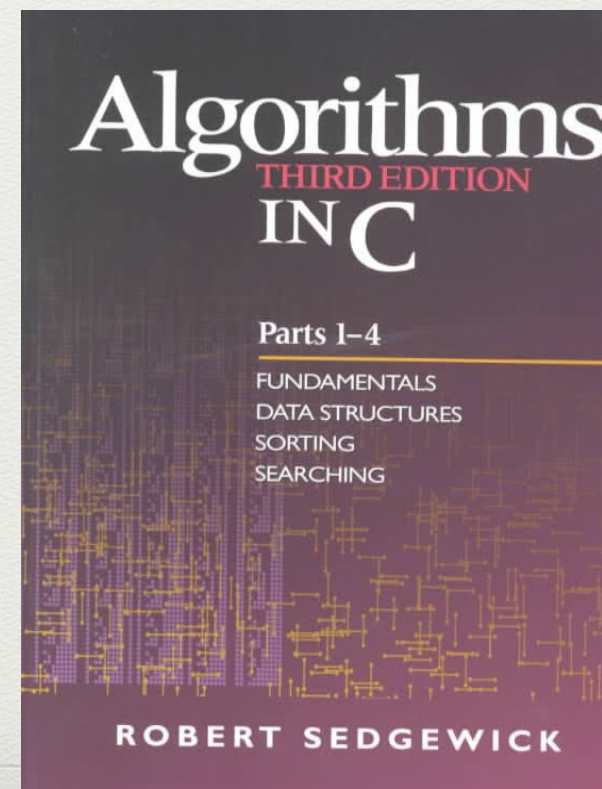
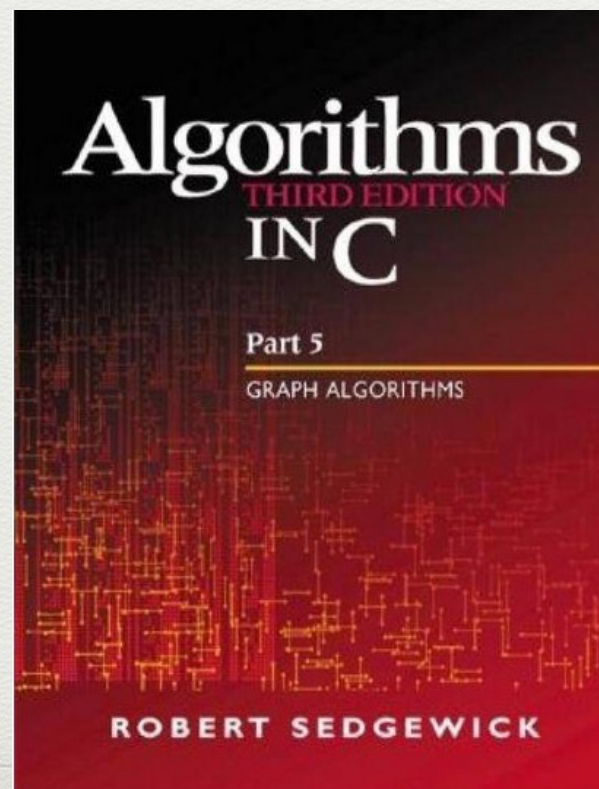
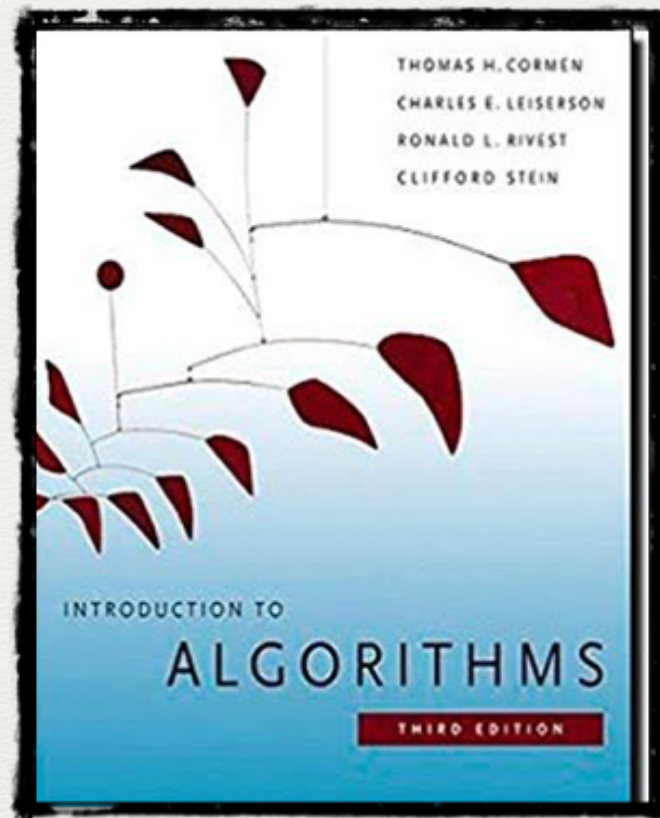
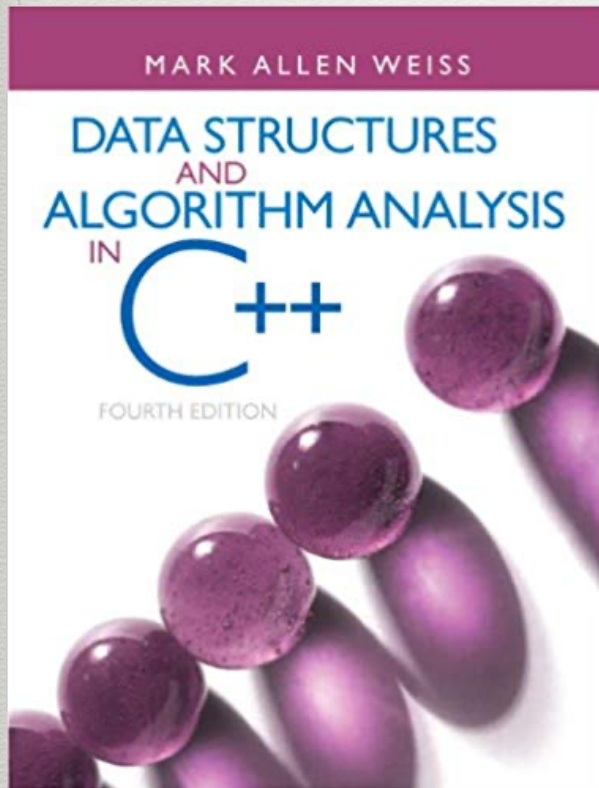


Classes Details

- Class A: Thursday, 09.20 – 11.05, Spada: SDA1A
- Class B: Thursday, 07.30 – 09.15, Spada: SDA1B
- Class C: Wednesday, 11.10 – 12.50, Spada: SDA1C
- The lab activities will be started in week 3 or 4
- All classes will have their own class coordinator and class secretary

Text Books



Class Rules

- No tolerance for cheating, plagiarism or any other type of dishonesty
 - **Consequency** —> **final grade: E**
- A student will only be considered present in the attendance report if he/she has entered the classroom at least 15 minutes after the lecturer enters the class. If you late more than one session (50 mins) then you're not allowed to enter the class
- A student who misses an examination must provide:
 - A verification of illness form indicating a severe illness, or
 - Other formal documentation, as appropriate

- All laptop computers, cell phones, tablet computers **must be closed** during all classroom hours.
 - If you wish to use a computer, you are welcome to step outside
- No late submissions will be allowed
 - Extensions may be allowed but only under extraordinary circumstances upon contacting me well in advance (generally at least 1 week prior to the submission date)
- Be attention! You only have less than 25% from all class sessions to absent.
- You may present to the other class in order to make up your class session, but you have to make a minimum of one-day notice before your absence.
- If you have any special condition, please let me know at least 3 weeks after the class started

Topics (only in Session 1)

- Introduction
- Asymtotic Analysis
- List, Stack and Queue
- Set, Map
- Tree
- Binary Search Tree
- AVL Tree
- Red Black Tree
- Hashing

Schedule

- W5: Mid Exam I
- W10: Mid Exam II

Assessments

Session II: 35

Session I: 65 → 105

- Mid Exam I: 35
- Mid Exam II: 35
- Assignments: 20
- Lab Attendances: 10
- Bonus: 5 —> participations in the class and exercises

Total: 105

Programming Language

- C
- C++

C++ can be said a superset of C. Major added features in C++ are Object-Oriented Programming, Exception Handling and rich C++ Library

Similarities between C and C++

- Both the languages have a similar syntax.
- Code structure of both the languages are same.
- The compilation of both the languages is similar.
- They share the same basic syntax. Nearly all of C's operators and keywords are also present in C++ and do the same thing.
- C++ has a slightly extended grammar than C, but the basic grammar is the same.
- Basic memory model of both is very close to the hardware.
- Same notions of stack, heap, file-scope and static variables are present in both the languages.

Differences between C and C++

C	C++
C was developed by Dennis Ritchie between the year 1969 and 1973 at AT&T Bell Labs.	C++ was developed by Bjarne Stroustrup in 1979.
C does not support polymorphism, encapsulation, and inheritance which means that C does not support object oriented programming.	C++ supports polymorphism , encapsulation , and inheritance because it is an object oriented programming language.
C is a subset of C++.	C++ is a superset of C.
C contains 32 keywords .	C++ contains 52 keywords .
For the development of code, C supports procedural programming .	C++ is known as hybrid language because C++ supports both procedural and object oriented programming paradigms .
Data and functions are separated in C because it is a procedural programming language.	Data and functions are encapsulated together in form of an object in C++.
C does not support information hiding.	Data is hidden by the Encapsulation to ensure that data structures and operators are used as intended.

C	C++
Built-in data types is supported in C.	Built-in & user-defined data types is supported in C++.
C is a function driven language because C is a procedural programming language.	C++ is an object driven language because it is an object oriented programming.
Function and operator overloading is not supported in C.	Function and operator overloading is supported by C++.
C is a function-driven language.	C++ is an object-driven language
Functions in C are not defined inside structures.	Functions can be used inside a structure in C++.
Namespace features are not present inside the C.	Namespace is used by C++, which avoid name collisions.
Header file used by C is stdio.h .	Header file used by C++ is iostream.h .
Reference variables are not supported by C.	Reference variables are supported by C++.
Virtual and friend functions are not supported by C.	Virtual and friend functions are supported by C++.

source: www.geeksforgeeks.com