

CS2303: DATA STRUCTURES FOR MEDIA

Effective Term

Semester A 2024/25

Part I Course Overview

Course Title

Data Structures for Media

Subject Code

CS - Computer Science

Course Number

2303

Academic Unit

Computer Science (CS)

College/School

College of Engineering (EG)

Course Duration

One Semester

Credit Units

3

Level

B1, B2, B3, B4 - Bachelor's Degree

Medium of Instruction

English

Medium of Assessment

English

Prerequisites

CS2313 Computer Programming or equivalent

Precursors

Nil

Equivalent Courses

Nil

Exclusive Courses

Nil

Part II Course Details

Abstract

This course aims to introduce a number of data structures and the mathematical tools for analysing their performance. Data structures that are commonly used for media are emphasised.

Course Intended Learning Outcomes (CILOs)

CILOs	Weighting (if app.)	DEC-A1	DEC-A2	DEC-A3
1	Evaluate and choose the appropriate data structures to solve problems.		x	
2	Explain, analyse, and compare diverse data structures.	x	x	
3	Design interactive games using suitably adapted data structures and apply specific data structures for media needs.			x

A1: Attitude

Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

A2: Ability

Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to real-life problems.

A3: Accomplishments

Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

Learning and Teaching Activities (LTAs)

LTAs	Brief Description	CILO No.	Hours/week (if applicable)	
1	Lecture	Students will explain the key concepts about algorithms and data structures and abstract data types, e.g., program complexities, searching, indexing, sorting, and manipulating data.	1, 2, 3	3 hours/week
2	Tutorial	Students will work on hands-on exercises (e.g., practice questions and coding tasks) and labs related to the key concepts and method covered in lectures.	2, 3	8 hours/semester

3	Project	Students will work on one project based on the knowledge learned in this course. The project is either an implementation of some complicated data structures taught in the course, or a study on an advanced data structure not covered in class. The students are also required to analyse the complexities of data structures and demonstrate their abilities to apply knowledge learned to solve new problems. The project should be documented in a project report.	2, 3	After class
4	Assignment	Students will be required to solve more challenging and practical problems which are about the key concepts and methods covered in the lecture. There will be 2 assignments in total, covering different key concepts covered in the course.	1, 2, 3	After class

Assessment Tasks / Activities (ATs)

	ATs	CILO No.	Weighting (%)	Remarks (e.g. Parameter for GenAI use)
1	Project	2, 3	15	
2	Assignments	1, 2, 3	10	
3	Tutorial coding tasks	2, 3	20	
4	Quiz	1, 2	5	1 mid-term quiz

Continuous Assessment (%)

50

Examination (%)

50

Examination Duration (Hours)

2

Additional Information for ATs

For a student to pass the course, at least 30% of the maximum mark for the examination must be obtained.

Assessment Rubrics (AR)

Assessment Task

Project

Criterion

ABILITY to DESIGN attractive games using suitable data structures

Excellent (A+, A, A-)

High

Good (B+, B, B-)

Significant

Fair (C+, C, C-)

Moderate

Marginal (D)

Basic

Failure (F)

Not even reaching marginal levels

Assessment Task

Assignment

Criterion

ABILITY to solve basic problems related to different data structures covered in the lecture

Excellent (A+, A, A-)

High

Good (B+, B, B-)

Significant

Fair (C+, C, C-)

Moderate

Marginal (D)

Basic

Failure (F)

Not even reaching marginal levels

Assessment Task

Tutorial coding tasks

Criterion

ABILITY to write code to implement basic data structures covered in the lecture or solve related problems

Excellent (A+, A, A-)

High

Good (B+, B, B-)

Significant

Fair (C+, C, C-)

Moderate

Marginal (D)

Basic

Failure (F)

Not even reaching marginal levels

Assessment Task

Quiz

Criterion

ABILITY to solve problems for different basic data structure topics

Excellent (A+, A, A-)

High

Good (B+, B, B-)

Significant

Fair (C+, C, C-)

Moderate

Marginal (D)

Basic

Failure (F)

Not even reaching marginal levels

Assessment Task

Examination

Criterion

ABILITY to solve problems for different advanced and media related data structures

Excellent (A+, A, A-)

High

Good (B+, B, B-)

Significant

Fair (C+, C, C-)

Moderate

Marginal (D)

Basic

Failure (F)

Not even reaching marginal levels

Part III Other Information

Keyword Syllabus

Complexities of programs: notation, average and worst case analysis, complexities of common programming constructs; Linked lists; Trees; Game trees; Abstract data types: stacks, queues, heaps, disjoint sets; Hash tables; Representation of vectors and bitmaps; Quadtrees and Octrees; Geometric structures.

Syllabus

- Program complexities
Asymptotic notations for program complexities. Complexities of common programming constructs, e.g., loops and recursive programs. Average and worst case analysis.
- Dynamic data structures
Linked list. Trees: Binary tree, Binary search trees. Balanced search trees.
- Abstract data types
Principles of abstract data types. Examples : stacks, queues, heaps.
- Hash tables
Direct addressing. Hash functions. Collision resolution.
- Vectors and Bitmaps
Representation of vector and bitmap data
- Quadtrees and Octrees
Structures of Quadtrees and Octrees and their uses in handling 2D and 3D data
- Geometric structures
Spatial layout and shape of geometric components and attributes; Connectivity of components

Reading List

Compulsory Readings

Title	
1	Nil

Additional Readings

Title	
1	Weiss M. (2013). Data Structures & Algorithm Analysis in C++. Addison Wesley, 4th edition.
2	Foley, van Dam, Feiner, Hughes (2013). Computer Graphics: Principles and Practice. Addison Wesley, 3rd edition.
3	http://site.ebrary.com/lib/cityu/detail.action?docID=10053633