Xi'an Jiaotong-Liverpool University
西交利物浦大学

DEPARTMENT OF INTELLIGENT SCIENCE

MODULE HANDBOOK

INT202
Complexity of Algorithms

RUI YANG

SEMESTER 2

2020-2021

SECTION A: Basic Information

- **Brief Introduction to the Module**

This module provides students with a comprehensive introduction to the modern study of computer algorithms. It presents basic data structures like queue, stack, AVL tree, and covers various algorithms in considerable depth such as sorting, encryption, greedy algorithm, dynamic programming, advanced graph algorithms, etc. The module prepares students with ability to design algorithm and analyse the algorithm complexity.

- **Key Module Information**

Module name:	Complexity of Algorithms
Module code:	INT202
Credit value:	5
Semester in which the module is taught:	Semester 2
Pre-requisites needed for the module:	None
Programmes on which the module is shared:	BSc ICS

- **Delivery Schedule**

Lecture/tutorial room: SC176
Lecture times: Wednesday 11:00-13:00
Tutorial times: Friday 9:00-11:00

- **Module Leader and Contact Details**

Name: Rui Yang
Email address: r.yang@xjtlu.edu.cn
Office telephone number: 0512-88161502
Room number and office hours: SD529 Thursday 11:00-13:00
Preferred means of contact: Email

SECTION B: What you can expect from the module

- **Educational Aims of the Module**

To demonstrate how the study of algorithmics has been applied in a number of different domains.
To introduce formal concepts of measures of complexity and algorithms analysis.
To introduce fundamental methods in data structures and algorithms design.
To make students aware of computationally hard problems and possible ways of coping with them.

- **Learning Outcomes**

On successful completion of the module, students are expected

1. To have an appreciation of the diversity of computational fields to which algorithmics has made significant contributions;
2. To have fluency in using basic data structures (queues, stacks, trees, graphs, etc) in conjunction with classical algorithmic problems (searching, sorting, graph algorithms, security issues) and be aware of basic number theory applications, etc.;
3. To be familiar with formal theories providing evidence that many important computational problems are inherently intractable, e.g., NP-completeness;
4. To be able to apply the knowledge gained in this module to the specification and analysis of data structures and algorithms.

- **Assessment Details**

Assessment 1: In-Class Test

This assessment aims to test learning outcomes 1 and 2. The assessment contributes 5% of the overall module grade. The in-class test will be arranged in Week 8.

Assessment 2: In-Class Test

This assessment aims to test learning outcomes 3 and 4. The assessment contributes 5% of the overall module grade. The in-class test will be arranged in Week 14.

Assessment 3: Problem Solving

This assessment aims to test learning outcomes 1, 2, 3 and 4. For the problems, besides the answers, you should also show how you get to it. A full mark will only be awarded for a correct answer with adequate explanations. The assessment contributes 10% of the overall module grade. Coursework must be submitted no later than 17:00 on June 6th 2021. An electronic copy should be uploaded to Learning Mall.

- **Methods of Learning and Teaching**

Students will be expected to attend three hours of formal lectures as well as to participate in one hour of supervised problem classes in a typical week. Lectures will introduce students to the academic content and practical skills which are the subject of the module, while problem classes will allow students to practice those skills.

In addition, students will be expected to devote three hours of unsupervised time to solving continuous assessment tasks and private study. Private study will provide time for reflection and consideration of lecture material and background reading.

Continuous assessment will be used to test to what extent practical skills have been learnt. A written examination at the end of the module will assess the academic achievement of students.

- **Syllabus & Teaching Plan**

Week Number	Lecture
Week 1	Introduction to module
Week 2	Basic Data Structures
Week 3	Linked List
Week 4	Trees
Week 5	Search Algorithm
Week 6	Sorting Algorithm
Week 8	Fundamental Techniques
Week 9	Graph
Week 10	Flow Matching
Week 11	Number Theory
Week 12	Number Theory and RSA
Week 13	NP-Complete Problems
Week 14	Review

- **Tutorial Schedule**

Student Group	Time	Day	Venue	Lecturer/Instructor
Group 1	9:00-11:00	Friday	SC176	Rui Yang

- **Reading Materials**

Reference Textbooks

Title	Author	ISBN/Publisher
Algorithm Design	M. T. Goodrich and R. Tamassia	9780471383659 / Wiley
Introduction To Algorithms	T. H. Cormen, C. E. Leiserson and R. L. Rivest	9780262530910 / MIT Press

SECTION C: Additional Information

- **Attendance**

Students who are able to be on campus are reminded of the Academic Policy requiring no less than 80% attendance at classes. Failure to observe this requirement may lead to failure or exclusion from resit examinations or retake examinations in the following year.

- **Student Feedback**

  The University is keen to elicit student feedback to make improvements for each module in every session. It is the University policy that the preferred way of achieving this is by means of an Online Module Evaluation Questionnaire Survey. Students will be invited to complete the questionnaire survey for this module at the end of the semester.

  You are strongly advised to read the policies mentioned below very carefully, which will help you better perform in your academic studies. All the policies and regulations related to your academic study can be found in 'Student Academic Services' section under the heading "Policies and Regulations" on E-bridge.

- **Plagiarism, Cheating, and Fabrication of Data.**

  Offences of this type can result in attendance at a University-level committee and penalties being imposed. You need to be familiar with the rules. Please see the "Academic Integrity Policy" available on e-Bridge in the 'Student Academic Services' section under the heading 'Policies and Regulations'.

- **Rules of submission for assessed coursework**

  The University has detailed rules and procedures governing the submission of assessed coursework. You need to be familiar with them. Details can be found in the "Code of Practice for Assessment" available on e-Bridge in the 'Student Academic Services' section under the heading 'Policies and Regulations'.

- **Late Submission of Assessed Coursework**

  The University attaches penalties to the late submission of assessed coursework. You need to be familiar with the University's rules. Details can be found in the "Code of Practice for Assessment" available on e-Bridge in the 'Student Academic Services' section under the heading 'Policies and Regulations'.

- **Mitigating Circumstances**

  The University is able to take into account mitigating circumstances, such as illness or personal circumstances which may have adversely affected student performance on a module. It is the student's responsibility to keep their Academic Advisor, Programme Director, or Head of Department informed of illness and other factors affecting their progress during the year and especially during the examination period. Students who believe that their performance on an examination or assessed coursework may have been impaired by illness, or other exceptional circumstances should follow the procedures set out in the "Mitigating

Circumstances Policy", which can be found on e-Bridge in the 'Student Academic Services' section under the heading 'Policies and Regulations'.

- **Learning Mall**

  Copies of lecture notes and other materials are available electronically through Learning Mall, the University's virtual learning environment.