UNIVERSITY OF MACAU
FACULTY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE
CISB210 Algorithm and Data Structures II
Syllabus
1st Semester 2014/2015
Part A – Course Outline

Compulsory course in Computer Science

Catalog description:
(2-2) 3 hours credit. Advanced data structures and algorithms. Algorithm Design Techniques. Introduction to NP problem.

Course type:
Theoretical with substantial laboratory/practice content.

Prerequisites:
- CISB120

Textbook(s) and other required material:
- Mark Allen Weiss, *Data Structures and Algorithm Analysis*, 2nd Edition, Addison-Wesley 1997 (Required)

References:
- Robert L. Kruse and Alexander J. Ryba, *Data Structures and Program Design*, Prentice Hall 1998.

Major prerequisites by topic:
- Programming in high level language
- Application of mathematical principals to the analysis of computing problems.
- Discrete mathematics.
- Basic knowledge of data structures and algorithms

Course objectives:
- Introduce to students advanced data structures [a,c,e].
- Introduce to students efficient algorithms for more difficult problems [a,c,e]
- Enhance students abilities in algorithm analysis [a,c,e].
- Introduce to students advanced techniques for algorithm design [a,c,e].
- Further enhance students programming abilities [c,e]
- Introduce to students reasoning of algorithms [a] (not measured).
- Introduce to students theory of computation [a] (not measured).

Topics covered:
- Heaps (3 weeks)
- Advanced trees (AVL trees) (2 weeks)
- Sorting (3 weeks)
- Graphs (2 weeks)
- Algorithm design techniques (1 week)
- Introduction to theory of computation (1 week)

Class/laboratory schedule:

Timetabled work in hours per week			No of teaching weeks	Total hours	Total credits	No/Duration of exam papers
Lecture	Tutorial	Practice				
2	Nil	2	14	56	3	1 / 3 hours

Student study effort required:

Class contact:	
Lecture	28 hours
Practice	28 hours
Other study effort	
Self-study and assignment	42 hours
Total student study effort	98 hours

Student assessment:
Final assessment will be determined on the basis of:
Homework and quizzes 15%
Exams 85%

Course assessment:
The assessment of course objectives will be determined on the basis of:
- Homework, quizzes and exams
- Course evaluation

Course outline:

Weeks	Topic	Course work
1-3	**Priority Queues** Simple implementation by lists, binary heaps, D-heaps, leftist heaps.	Assignment#1
4-5	**Advanced Trees** AVL trees	Assignment#2
6	**Review and Middle Term Exam**	Middle Term Exam
7-9	**Sorting** Insertion sort, lower bound for simple sorting algorithms, shell sort, merge sort, quick sort, general lower bound for sorting based on comparison, external sorting.	Assignment#3
10-11	**Graph** Representation of graph, traversal, topological sort, shortest path, minimal spanning tree.	Assignment#4
12	**Techniques for Algorithm Design** Greedy, divide and conquer, dynamic programming, backtracking.	
13	**Introduction to Theory of Computation** Undecidable problems, tractable and untractable problems, NP problems, randomized algorithms.	
14	There are several holidays in the semester. This week will be flexible, and if there is time, will spend more time on algorithm design.	

Contribution of course to meet the professional component:
This course further prepares students to work professionally in the area of advanced programming.

Relationship to CS program objectives and outcomes:
This course primarily contributes to the Computer Science program outcomes that develop student abilities to:
(a) apply knowledge of computing, mathematics, science, and engineering
(c) design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
(e) an ability to identify, formulate, and solve engineering problems

.