Lovely Professional University, Punjab

Course Code	Course Title	Course Planner
CSE408	DESIGN AND ANALYSIS OF ALGORITHMS	11057::Gaurav Pushkarna

Course Outcomes :Through this course students should be able to

CO1 :: explain the basic techniques of analyzing the algorithms using space and time complexity, asymptotic notations

CO2 :: analyse various string matching algorithms and understand brute force algorithm design technique

CO3 :: understand divide and conquer algorithm design technique using various searching and sorting algorithms

CO4 :: define dynamic programming and greedy algorithm design technique and solve various all pair and single source shortest path problems

CO5 :: apply the backtracking method to solve some classic problems and understand branch and bound algorithm design technique

CO6 :: define various number theory problems and understand the basics concepts of complexity classes

	TextBooks (T)		
Sr No	**Title**	**Author**	**Publisher Name**
T-1	INTRODUCTION TO THE DESIGN AND ANALYSIS OF ALGORITHM	ANANY LEVITIN	PEARSON

	Reference Books (R)		
Sr No	**Title**	**Author**	**Publisher Name**
R-1	INTRODUCTION TO ALGORITHMS	C.E. LEISERSON, R.L. RIVEST AND C. STEIN	THOMAS TELFORD LTD.
R-2	THE DESIGN AND ANALYSIS OF COMPUTER ALGORITHMS	A.V.AHO, J.E. HOPCROFT AND J.D.ULLMAN	PEARSON
R-3	COMPUTER ALGORITHMS - INTRODUCTION TO DESIGN AND ANALYSIS	SARA BAASE AND ALLEN VAN GELDER	PEARSON
R-4	FUNDAMENTALS OF COMPUTER ALGORITHMS	HOROWITZ, S. SAHNI	GALGOTIA PUBLICATIONS

Other Reading (OR)	
Sr No	**Journals articles as Compulsary reading (specific articles, complete reference)**
OR-1	http://www.cs.cornell.edu/~kozen/papers/daa.pdf ,

OR-2	http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2012/lecture-notes/ ,
OR-3	http://cgi.csc.liv.ac.uk/~ped/teachadmin/algor/algor_complete.html ,
OR-4	http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html ,

Relevant Websites (RW)

Sr No	(Web address) (only if relevant to the course)	Salient Features
RW-1	http://nptel.ac.in/courses/106101060/	Asymptotic Notations,Divide And Conquer,Dynamic Programming
RW-2	http://www.cse.yorku.ca/~aaw/Zambito/TSP_L/Web/TSPStart.html	Travelling Salesman Problem
RW-3	http://optlab-server.sce.carleton.ca/POAnimations2007/DijkstrasAlgo.html	Dijkstra's shortest path
RW-4	http://www.wiwi.uni-jena.de/entscheidung/binpp/	Bin Packing Problems
RW-5	https://courses.ncsu.edu/ma103/common/media/05/MA103Lct25.mp4	Prims and Kruskals algorithms
RW-6	http://www3.cs.stonybrook.edu/~algorith/files/dfs-bfs.shtml	Connected Components
RW-7	http://mathworld.wolfram.com/VoronoiDiagram.html	Voronoi Diagram
RW-8	http://math.mit.edu/~stevenj/18.335/optimization-handout.pdf	Optimization Problems

Audio Visual Aids (AV)

Sr No	(AV aids) (only if relevant to the course)	Salient Features
AV-1	https://www.cs.usfca.edu/~galles/visualization/ComparisonSort.html	Comparison Sorting Algorithms

LTP week distribution: (LTP Weeks)	
Weeks before MTE	7
Weeks After MTE	7
Spill Over (Lecture)	7

Detailed Plan For Lectures

Week Number	Lecture Number	Broad Topic(Sub Topic)	Chapters/Sections of Text/reference books	Other Readings, Relevant Websites, Audio Visual Aids, software and Virtual Labs	Lecture Description	Learning Outcomes	Pedagogical Tool Demonstration/ Case Study / Images / animation / ppt etc. Planned	Live Examples

An instruction plan is only a tentative plan. The teacher may make some changes in his/her teaching plan. The students are advised to use syllabus for preparation of all examinations. The students are expected to keep themselves updated on the contemporary issues related to the course. Upto 20% of the questions in any examination/Academic tasks can be asked from such issues even if not explicitly mentioned in the instruction plan.

Week 1	Lecture 1	Foundations of Algorithm (Algorithms)	T-1	OR-4 RW-7	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Introduction of algorithms,the structure of algorithms	Student will learn the need of algorithms for real world applications	Demonstration with power point presentation	Solution to given problem
		Foundations of Algorithm (Fundamentals of Algorithmic Problem Solving:)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Introduction of algorithms,the structure of algorithms	Student will learn to solve a problem and write in algorithmic form	Demonstration with power point presentation	Solution to given problem
		Foundations of Algorithm (Basic Algorithm Design Techniques)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Design techniques(Divide and conquer,Greedy,Dynamic,Back tracking,Branch and bound)	Student will learn the usage of Design techniques and to apply them to get the solution for any problem	Demonstration with power point presentation	
		Foundations of Algorithm (Analyzing Algorithm)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:A prior analysis and posterior analysis	Student will learn to choose a prior or posterior to analyze any problem	Demonstration with power point presentation	Choosing the best from available after analysis
		Foundations of Algorithm (Fundamental Data Structure:)	T-1		Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Introduction of data structures(Linear and non Linear)	Student will learn the usage of Data Structure to store the data	Demonstration with power point presentation	Solution to given problem

Week 1	Lecture 1	Foundations of Algorithm (Linear Data Structure)	T-1		Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Description of arrays linked list stack and ques	Student will learn the usage of linear data structures to store the data	Demonstration with power point presentation	Students sitting in a class room in series,pile of plates,ticket counter
		Foundations of Algorithm (Graphs and Trees)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Description of Non linear Data structure(Tree,Graph)	Student will learn the usage of Non linear data structure to design the algorithm	Demonstration with power point presentation	Cities connected with roads
	Lecture 2	Foundations of Algorithm (Algorithms)	T-1	OR-4 RW-7	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Introduction of algorithms,the structure of algorithms	Student will learn the need of algorithms for real world applications	Demonstration with power point presentation	Solution to given problem
		Foundations of Algorithm (Fundamentals of Algorithmic Problem Solving:)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Introduction of algorithms,the structure of algorithms	Student will learn to solve a problem and write in algorithmic form	Demonstration with power point presentation	Solution to given problem
		Foundations of Algorithm (Basic Algorithm Design Techniques)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Design techniques(Divide and conquer,Greedy,Dynamic,Back tracking,Branch and bound)	Student will learn the usage of Design techniques and to apply them to get the solution for any problem	Demonstration with power point presentation	

Week 1	Lecture 2	Foundations of Algorithm (Analyzing Algorithm)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:A prior analysis and posterior analysis	Student will learn to choose a prior or posterior to analyze any problem	Demonstration with power point presentation	Choosing the best from available after analysis
		Foundations of Algorithm (Fundamental Data Structure:)	T-1		Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Introduction of data structures(Linear and non Linear)	Student will learn the usage of Data Structure to store the data	Demonstration with power point presentation	Solution to given problem
		Foundations of Algorithm (Linear Data Structure)	T-1		Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Description of arrays linked list stack and ques	Student will learn the usage of linear data structures to store the data	Demonstration with power point presentation	Students sitting in a class room in series,pile of plates,ticket counter
		Foundations of Algorithm (Graphs and Trees)	T-1	OR-4	Lecture1:Zero lecture for the Introduction of course ,objectives,structures and details of academic tasks Lecture2:Description of Non linear Data structure(Tree,Graph)	Student will learn the usage of Non linear data structure to design the algorithm	Demonstration with power point presentation	Cities connected with roads
	Lecture 3	Foundations of Algorithm (Fundamentals of the Analysis of Algorithm Efficiency:)	T-1 R-4		Basic knowledge about concepts of complexities of algorithms,its definition	Student will learn to find the best algorithm after analysis	mathematical expressions will be used to illustrate the concept	
		Foundations of Algorithm (Measuring of Input Size)	T-1		Different type of input sets	Student will learn to differentiate different input sets	mathematical expressions will be used to illustrate the concept	

Week 1	Lecture 3	Foundations of Algorithm (Units for Measuring Running Time)	T-1		Different notations to measure running time and its definition	Student will learn to find the running time of an algorithm	mathematical expressions will be used to illustrate the concept	
		Foundations of Algorithm (Order of Growth)	T-1		Order of growth ,different timings	Student will learn to find order of growth (time) for given algorithm	mathematical expressions will be used to illustrate the concept	
		Foundations of Algorithm (Worst-Case, Best-Case, and Average-Case Efficiencies)	T-1		Different time complexities (worst,best,avg)	Student will learn to differentiate between cases and their usage	mathematical expressions will be used to illustrate the concept	Time taken to run the program ,min ,max
Week 2	Lecture 4	Foundations of Algorithm (Asymptotic Notations and Basic Efficiency Classes:)	T-1 R-4	OR-3	Asymptotic notations (Lower bound,Upper bound,Tight bound)	Student will learn to represent the time and space complexity using notations	relevant graphical representation of every case to be used	
		Foundations of Algorithm(O (Big-oh)-notation, Big-omega notation, Big-theta notation)	T-1		Lower bound,Upper bound,Tight bound	Student will learn the usage of notations to represent the complexity for any algorithm	relevant graphical representation of every case to be used	Comparison of upper and lower floors of building for upper bound and lower bound
		Foundations of Algorithm (Useful Property Involving the Asymptotic Notations)	T-1		The properties of asymptotic notations	Student will learn the usage of properties to represent the complexity for any algorithm	relevant graphical representation of every case to be used	
		Foundations of Algorithm (Using Limits for Comparing Orders of Growth)	T-1		Significance of limits to compare orders of growth	Student will learn the usage of order of growth for any algorithm	relevant graphical representation of every case to be used	
	Lecture 5	String Matching Algorithms and Computational Geometry(Naiva String-Matching Algorithm)	R-1		Introduction to String matching,NSM algo,and example on NSM	Student will learn the usage of NSM to search the data from huge data base	Numerical problem solving	DNA search
		String Matching Algorithms and Computational Geometry(Rabin-Karp Algorithm)	R-1		Robin karp algo,definition,Modules operation.	Students will learn to search pattern from given text for integers	Numerical problem solving	DNA search
	Lecture 6	String Matching Algorithms and Computational Geometry(Knuth-Morris-Pratt Algorithm)	R-1 R-3		KMP algo,definition,prefix computation	Student will learn to search the pattern from text	Numerical	DNA search
Week 3	Lecture 7	String Matching Algorithms and Computational Geometry(Sequential Search and Brute-Force String Matching)	T-1		Definition of brute force,Brute force string matching	Student will learn to search the pattern form given text.	Numerical Problem Solving	DNA search

Week 3	Lecture 7	String Matching Algorithms and Computational Geometry(Closest-Pair and Convex-Hull Problem)	T-1		Closest pair and definition of convex hull and to get closest pair,convex hull	Student will learn to find closest pair and making of convex hull.	Numerical Problem Solving	Finding smallest distance among adjacent cities
	Lecture 8	String Matching Algorithms and Computational Geometry(Exhaustive Search)	T-1		Definition of exhaustive search with examples	Student will learn to search the elements till end	Numerical Problem Solving	
	Lecture 9	Divide and Conquer and Order Statistics(Merge Sort and Quick Sort)	T-1 R-3	RW-1	Introduction of divide and conquer method and Merge sort and quick sort	Student will learn the usage of divide and conquer to solve the merge and quick sort	Numerical Problem Solving	Combine the answer sheets after sorting packet wise
Week 4	Lecture 10	Divide and Conquer and Order Statistics(Binary Search)	T-1		Different types of Divide and Conquer and Order Statistics techniques	Student will learn the reduction of complexity to search elements over linear search	Numerical Problem Solving	
		Divide and Conquer and Order Statistics (Multiplication of Large Integers)	T-1		Different types of Divide and Conquer and Order Statistics techniques,multiplication of large integers.	Student will learn the usage of divide and conquer to reduce the complexity for multiplication of large integers	Numerical Problem Solving	
		Divide and Conquer and Order Statistics(Strassen's Matrix Multiplication)	T-1 R-2		Different types of Divide and Conquer and Order Statistics techniques,strassen's matrix multiplication	Student will learn the importance of divide and conquer to reduce the complexity of strassen's matrix multiplication over classical multiplication	Numerical Problem Solving	
	Lecture 11				Online Assignment			
	Lecture 12	Divide and Conquer and Order Statistics(Substitution Method for Solving Recurrences)	T-1 R-1		Substitution Method,its definition and procedure	Students will learn the significance of substitution method and to find the time complexity	Numerical problem solving	
		Divide and Conquer and Order Statistics(Recursion-Tree Method for Solving Recurrences)	T-1 R-1		To find the time complexities for recurrence algorithms	Students learn the significance of recurrence tree method and how to find time complexity	Numerical problem solving	

Week 4	Lecture 12	Divide and Conquer and Order Statistics(Master Method for Solving Recurrence)	T-1 R-1		Master method,its definition along with three cases.	Student will learn the significance of master method and the way it helps to solve the recurrence equation	Numerical problem solving	
Week 5	Lecture 13	Divide and Conquer and Order Statistics(Closest-Pair and Convex-Hull Problems by Divide and Conquer)	T-1		Closest pair and convex hull ,its definitions along with procedures	Student will learn to find the two closest pairs and to make the convex hull	Numerical problem solving	Nearest tow adjacent cities
		Divide and Conquer and Order Statistics(Decrease and Conquer: Insertion Sort)	T-1		Insertion sort,its definition along with algorithm and examples	Student will learn the usage of decrease and conquer to sort the elements with insertion sort	Numerical problem solving	
	Lecture 14	Divide and Conquer and Order Statistics(Depth-First Search and Breadth-First Search)	T-1	OR-1	Graph traversal, BFS,DFS,algorithms of BFS,DFS	Student will learn to traverse from one node to another node,to cover all node	paper referred will be used to explain the concept	Traveling all cities existed in a state
		Divide and Conquer and Order Statistics(Connected Components)	T-1	RW-6	Connected components,to find connected components from the graph	Student will learn the usage of connected components for locating the clusters in graph	paper referred will be used to explain the concept	
	Lecture 15	Divide and Conquer and Order Statistics(Transform and Conquer: Presorting)	T-1		Transform and conquer ,presorting, its definition	Student will learn the usage transform and conquer for presorting	relevant graphical representation to be used	
		Divide and Conquer and Order Statistics(Balanced Search Trees)	T-1		Balanced search trees (AVL and 2-3 tree)	Student will learn the usage of Balanced search tree to reduce the complexity over Binary search tree	Animation	Family structure
Week 6	Lecture 16	Divide and Conquer and Order Statistics(Minimum and Maximum)	T-1		To get the min and maximum elements with procedure	Student will learn to get min and max elements form given elements	Numerical problem solving	
		Divide and Conquer and Order Statistics(Selection Sort and Bubble Sort)	T-1	AV-1	Sorting techniques (Selection and bubble sort)	Student will learn the usage of sorting to sort the elements	link referred will be used to demonstrate the concept using AV aid	telephone directory
	Lecture 17	Divide and Conquer and Order Statistics(Counting Sort, Radix Sort, Bucket Sort)	T-1	AV-1	Sorting techniques (Counting ,Radix,Bucket sort) with examples	Student will learn to sort elements using different techniques to reduce the time taking	link referred will be used to demonstrate the concept using AV aid	telephone directory

Week 6	Lecture 18	Divide and Conquer and Order Statistics(Heaps and Heapsort)	T-1	AV-1	Hashing(division,mid square,folding) Creation of heap,deletion and heap sort.	Student will learn the usage of hashing to search the element with order of constant Student will learn the usage of heap to reduce the time to sort the element	numerical problem solving link referred will be used to demonstrate the concept using AV aid	
		Divide and Conquer and Order Statistics(Hashing)	T-1	AV-1	Hashing(division,mid square,folding) Creation of heap,deletion and heap sort.	Student will learn the usage of hashing to search the element with order of constant Student will learn the usage of heap to reduce the time to sort the element	numerical problem solving link referred will be used to demonstrate the concept using AV aid	
Week 7	Lecture 19				Online Assignment			
		SPILL OVER						
Week 7	Lecture 20				Spill Over			
	Lecture 21				Spill Over			
		MID-TERM						
Week 8	Lecture 22	Dynamic Programming and Greedy Techniques (Dynamic Programming: Computing a Binomial Coefficient)	T-1		Introduction of dynamic programming ,Binomial cofficient	Student will learn use of dynamic programming to find binomial coefficients	Numerical Problem Solving	
		Dynamic Programming and Greedy Techniques (Warshall's and Floyd's Algorithm)	T-1		Working of Warshal and floyds algorithms with examples	Student will learn to find all pairs shortest path using dynamic programming approach	Numerical Problem Solving	Finding shortest distance between all cities existed in a state
	Lecture 23	Dynamic Programming and Greedy Techniques(Optimal Binary Search Trees)	T-1		Optimal binary search tree with dynamic approach	Student will learn to find optimal BST from given BST'S	numerical problem solving	
		Dynamic Programming and Greedy Techniques (Knapsack Problem and Memory Functions)	T-1	RW-1	Knapsack problem,its definition ,dynamic approach	Student will learn the usage of knapsack for e-marketing	referred link will be used to explain the concept	
	Lecture 24	Dynamic Programming and Greedy Techniques(Matrix-Chain Multiplication)	R-1		Classic matrix multiplication, matrix chain multiplication	Student will learn the use of MCM to reduce the complexity over classical matrix multiplication	Numerical Problem Solving	

Week 9	Lecture 25	Dynamic Programming and Greedy Techniques(Longest Common Subsequence)	R-1		Definition of LCS,algorithm of LCS with example	Student will learn to get common sub sequence form two lists	Numerical Problem Solving	
	Lecture 26	Dynamic Programming and Greedy Techniques(Greedy Technique and Graph Algorithm: Minimum Spanning Trees)	T-1 R-1	OR-2	MST(Prim's and kruskal)	Student will learn to find the Shortest distance from one node to another node	classroom discussion	
		Dynamic Programming and Greedy Techniques(Prim's Algorithm)	T-1 R-1	RW-5	Different type of techniques to find out Minimum Spanning Trees	Student will learn to find the Minimum spanning tree with prim's algorithm	relevant graphical representation to elucidate the topic	
		Dynamic Programming and Greedy Techniques (Kruskal's Algorithm)	T-1 R-1	RW-5	Different type of techniques to find out Minimum Spanning Trees	Students will learn to find out Minimum Spanning Trees using kruskal algorithm	relevant graphical representation to elucidate the topic	
	Lecture 27	Dynamic Programming and Greedy Techniques (Dijkstra's Algorithm)	T-1	RW-3	Single source shortest path ,Dijkstra algorithm with example	Student will learn to find out shortest path using Dijkstra algorithm	Numerical Problem Solving	
		Dynamic Programming and Greedy Techniques (Huffman Code)	T-1		Huffman coding,To find code for encryption	Student will learn to encrypt the text using huffman coding algorithm	Numerical Problem Solving	
Week 10	Lecture 28	Dynamic Programming and Greedy Techniques(Single-Source Shortest Paths)	R-1		Different type of Single Source Shortest Paths Problems	Student will learn to find the Single Source Shortest Paths from one node to all nodes	numerical problem solving	
	Lecture 29	Dynamic Programming and Greedy Techniques(All-Pairs Shortest Paths)	R-1		All pairs shortest path, Floyd-warshal with example	Student will learn to find the shortest distance form all nodes to all	numerical problem solving	
	Lecture 30	Dynamic Programming and Greedy Techniques(Iterative Improvement: The Maximum-Flow Problem)	T-1		Description of The Maximum Flow Problem and cut	student will learn the usage of maximum flow in networks	Numerical Problem Solving	
Week 11	Lecture 31	Dynamic Programming and Greedy Techniques (Limitations of Algorithm Power: Lower-Bound Theory)	T-1		Description of Lower Bound Theory	Student will the Learn Limitations of Algorithm Power	classroom discussion	
	Lecture 32	Backtracking and Approximation Algorithms (Backtracking: n-Queens Problem)	T-1		Description of Backtracking problem and n queens problem	Student will learn the usage of Back tracking to solve the n queen problem	Numerical Problem Solving	Maze

Week 11	Lecture 32	Backtracking and Approximation Algorithms (Hamiltonian Circuit Problem)	T-1		Description of Backtracking problem,Hamiltonian circuit problem	Student will learn the usage of backtracking to solve the Hamiltonian circuit problem	Numerical Problem Solving	
	Lecture 33				Online Assignment			
Week 12	Lecture 34	Backtracking and Approximation Algorithms (Subset-Sum Problem)	T-1		Backtracking,subset sum problem	Student will learn to find sum form subsets	Numerical Problem Solving	
		Backtracking and Approximation Algorithms (Branch-and-Bound: Assignment Problem)	T-1		Introduction of Branch and bound,Assignment problem	Student will learn to assign different jobs to different people	Numerical Problem Solving	
	Lecture 35	Backtracking and Approximation Algorithms (Knapsack Problem)	T-1		To place max number of items in knapsack to get max profit	Student will Learn to get max profit after placing items in knapsack	mathematical analysis	
	Lecture 36	Backtracking and Approximation Algorithms (Traveling Salesman Problem)	T-1	RW-2	Definition of TSP,Backtracking approach to solve TSP	Student will learn to find optimal value to traverse all the cities	Numerical Problem Solving	
Week 13	Lecture 37	Backtracking and Approximation Algorithms (Vertex-Cover Problem and Set-Covering Problem)	R-2 R-4		Vertex cover and set cover problems,its definitions with examples	Student will learn to cover all vertices from given	Numerical Problem Solving	
		Backtracking and Approximation Algorithms (Bin Packing Problems)	R-2 R-4	RW-4	Bin packing ,its definition,examples	Student will Learn to pack different sizes of items with Bins	Numerical Problem Solving	
	Lecture 38	Number-Theoretic Algorithms and Complexity Classes(Number Theory Problems: Modular Arithmetic)	R-1		Description of Number theory problems and Modular Arithmetic,their definitions	Student will learn the use of modular arithmetic for arithmetic operations	Numerical problem solving	
		Number-Theoretic Algorithms and Complexity Classes(Chinese Remainder Theorem)	R-1 R-2		Chinese remainder theorem,definition and theory of chinese remainder theorem	Students will learn to use the chinese remainder theorem for arithmetic operations	mathematical analysis	
	Lecture 39	Number-Theoretic Algorithms and Complexity Classes(Greatest Common Divisor)	R-1		GCD,ecluids algorithm	Student will learn to get the GCD of two numbers	Numerical problem solving	
		Number-Theoretic Algorithms and Complexity Classes(Optimization Problems)	R-1	RW-8	Optimization problem,its definition	Student will learn to get the gcd of two numbers	referred link will be used to explain the concept	

Week 14	Lecture 40	Number-Theoretic Algorithms and Complexity Classes(Basic Concepts of Complexity Classes- P, NP, NP-hard, NP-complete Problems)	T-1		P NP NP complete and NP hard problems	Student will learn the categorization of all the algorithms	classroom discussion	
		SPILL OVER						
Week 14	Lecture 41				Spill Over			
	Lecture 42				Spill Over			
Week 15	Lecture 43				Spill Over			
	Lecture 44				Spill Over			
	Lecture 45				Spill Over			