POSTGRADUATE INSTITUTE OF SCIENCE

UNIVERSITY OF PERADENIYA

Master of Computer Science Degree Programme
(SLQF Level 9)

Master of Science (M.Sc.) in Computer Science Degree Programme
(SLQF Level 10)

1. INTRODUCTION

Most of the science degree students require knowledge of Computer Science in order to find a suitable job. On the other hand, those who have done Computer Science as a subject during the undergraduate programme are looking for a competitive job. Information technology will exert a great influence in all science subjects in the future and therefore it is essential to train graduates for the next millennium. Computer science will be the first as well as the most innovative discipline that can interact with any other discipline in order to develop subjects, which have areas of interaction.

The knowledge of computer science is a must for undergraduate as well as postgraduate degree students. All the enterprises (institutes, organizations, and companies) should have at least one qualified computer scientist. In this decade, most institutes need to develop by introducing and/or improving computer science in their curricula to suit actual demands. The postgraduate degree programme envisaged will give a comprehensive knowledge of recent issues in computer science.

2. OBJECTIVES OF THE PROGRAMME

The objective of this programme is to provide advanced and enhanced knowledge of recent issues of information technology. At the completion of this course, the candidates will be able to fit into any computerized enterprise or a research institute.

3. PROGRAMME ELIGIBILITY

Applicants must possess a science-based degree (e.g. Physical/Biological/Engineering related degree*)* or any other degree (where a basic knowledge of Calculus, Number Theory and Algebra has been obtained) with 30 credits in the relevant subject area or an equivalent qualification acceptable to the Postgraduate Institute of Science. Graduates, who have no basic knowledge in computer science, are expected to follow fundamentals of computer science (None credit courses). Depending on the courses followed at the degree level and on the recommendation of the advisor a candidate may be exempted from some of the preliminary courses. Eligible applicants will face a selection examination,

conducted by the PGIS. Employed candidates who are eligible for admission should produce evidence of leave granted to follow the programme and a letter of release from the Head of the Department/Institution.

4. PROGRAMME FEE

Category	Programme Fee	
	Master of Computer Science degree programme	**M.Sc. in Computer Science degree programme**
Local candidates	Rs. 380,000/-	Rs. 480,000/-
Foreign candidates	Rs. 760,000/-	Rs. 960,000/-

Students registered for the Master of Computer Science degree programme shall pay the Programme fee in full or in two or in three installments. An additional payment of Rs. 100,000/- (or Rs. 200,000/- form foreign students) should be made at the end of the first year to continue for the M.Sc. in Computer Science degree programme. Other payments including registration fee, medical fee, library subscription, examination fee and deposits (science and library) should be paid according to the procedure stipulated by the PGIS. (N.B. The Programme fees given above may be revised as per recommendation of the Board of Management of the PGIS.)

5. THE PROGRAMME STRUCTURE AND DURATION

This programme consists of three options for completion.

5.1 Masters Degree by Course Work (SLQF Level 9)

The Master of Computer Science degree can be obtained by completing course work only (without conducting any research project).

Course work, comprising of theory courses, and laboratory and/or fieldwork, shall be conducted over a period of two semesters of 15 weeks each. The total duration of the degree, including examinations, shall be about 12 months. Satisfactory completion of a minimum of 30 credits of course work with a GPA of not less than 3.00 is required for the successful completion of the degree (The student who does not satisfy the above criteria but obtains a GPA in the range 2.75 to 2.99 for course work of 25 30 credits is eligible for the Diploma in Computer Science).

5.2 Masters Degree (SLQF Level 10)

In addition to Masters Degree with course work (5.1), the Masters Degree (Research) requires a research project. The duration of the entire programme will be 24 months inclusive of 5.1. Completion of all the requirements of 5.1 with a GPA of not less than 3.00 is a prerequisite for the Masters Degree (Research). The research project for this degree should be conducted on full-time basis, and completed during the second year. The research component is allocated 30 credits, totalling 60 credits for the entire programme. After successful completion of the research project, the student shall be eligible for the award of the M.Sc. in Computer Science degree - SLQF Level 10 (The student who does not complete the research project shall be awarded the Master of Computer Science degree - SLQF Level 9).

5.3 Extension of the programme for M.Phil. or Ph.D.

After completion of six months of research in the M.Sc. degree (research) programme, students who have demonstrated exceptional progress may apply for upgrading the degree status to M.Phil. The student should continue the research project and any additional research work/assignments recommended by the PGIS for a total of two years (60 credits of research) to qualify for the award of the M.Phil. degree.

During the second year of research, students who have demonstrated exceptional and continuous progress, may apply for upgrading the degree status from M.Phil. to Ph.D. The student should continue the research project and any additional research work/assignments recommended by the PGIS for another year on full-time basis (additional 30 credits) to qualify for the award of the Ph.D. degree.

Master of Computer Science Degree Programme (SLQF Level 9)
Master of Science (M.Sc.) in Computer Science Degree Programme (SLQF Level 10)

Programme Summary

Course Code	Course	Lecture hrs.	Practical hrs.	No. of Credits
Preliminary Courses				
SC 411	Introduction to Computer Science	30	-	-
SC 412	Introduction to Theory of Computation	30	-	-
SC 414	Introduction to Computer Architecture	30	-	-
SC 415	Programming and Electronics Laboratories	-	45	-
SC 416	Seminar	15	-	-
SC 417	Data Structures and Algorithms	30	-	-
SC 418	Programming Techniques	25	20	-
Semester I				
SC 531	Database Systems *	30	-	2
SC 532	Combinatorial Mathematics	30	-	2
SC 533	Introduction to Parallel Computing	30	-	2
SC 534	Programming Language Design and Compilers	30	-	2
SC 535	Operating System Design *	30	-	2
SC 536	Graph Theory	30	-	2
SC 537	Computer Networks & Distributed systems *	30	-	2

SC 538	Artificial Intelligence *	30	-	2
SC 539	Advanced Topics in Computer Graphics *	30	-	2
SC 540	Web Servers and Technologies	30		2
SC 541	Programming Web Applications		60	2
SC 542	Introduction to Wireless and Mobile Systems	30		2
SC 543	Object Oriented Analysis and Design	30		2
Semester II				
SC 546	Software Engineering *	30	-	2
SC 547	Computer Architecture	30	-	2
SC 548	Systems Analysis/Systems Engineering	30	-	2
SC 549	Artificial Neural Networks	30	-	2
SC 550	Linear Programming	30	-	2
SC 551	Communication Networks for Computers	30	-	2
SC 552	Digital Image Processing	30	-	2
SC 553	Project Management	30	-	2
SC 554	Special Topics in Computer Science *	30	-	2
SC 555	Laboratory Work *	-	60	2
SC 556	Computer Vision	30		2
SC 557	Analysis of Communication Networks	30		2
SC 558	Cryptography and Network Security	25	20	2
SC 559	Multimedia Systems	30		2
SC 560	Multimedia Practical		60	2
SC 561	Data Mining Techniques	30		2
SC 599	Independent Study *	500 notional hours		5
SC 699	Research Project (One Year Duration) **	3000 notional hours		30
	Total Credits			**93**

Preliminary courses are not considered in the computation of the GPA

* *Compulsory Courses*

** *Compulsory for M.Sc. in Computer Science (SLQF Level 10)*

PROGRAMME CONTENTS OF SC 599 AND SC 699

Course Code	SC 599
Course Title	Independent Study
Credits	5
Compulsory/Optional	Compulsory
Time allocation	500 notional hours
Aims	The overall aim is to familiarize the student with concepts and methods involved in scientific research. Specific aims: 1. To learn the scientific process in the conduct of research. 2. To develop skills to write a review paper and a scientific research proposal. 3. To develop skills to make a presentation. 4. To carry out a case study in a Computer Science/Statistics related problem
Intended Learning Outcomes	At the end of the successful completion of the course module, students will be able to, 1. Conduct an independent review of literature on a selected topic in the area of Computer Science/Statistics. 2. Write a formal scientific report conforming to the guidelines provided. 3. Complete a research proposal conforming to the guidelines provided. 4. Carry out a mini research project or a case study on a Computer Science/Statistics related problem. 5. Transfer the knowledge gained through (1) and (4) above in the form of a presentation.
Content	*Review paper*: Review of literature; Development of the review paper in concise and professional manner and logical presentation of results that have been reported, writing the abstract, compilation of the list of references. *Proposal writing*: Interpretation and critical evaluation of results of published research; Formulation of a research problem: Concise literature review, justification, time frame, identification of resources, budgeting, etc. *Seminar*: Presentation of literature and data collected on a given topic; Preparation of an abstract, preparation of slides. *Case Study*: Presentation of the results and conclusions of a case study.

Assessment criteria: Continuous Assessment

Component	Review Paper	Proposal Writing	Seminar	Case Study
%marks	20%	10%	30%	40%

Recommended Texts:

1. Backwell, J., Martin, J. (2011) *A Scientific Approach to Scientific Writing*, Springer.
2. Postgraduate Institute of Science (2016) *Guidelines for Writing M.Sc. Project Report/M.Phil. Thesis/Ph.D. Thesis*

Course code	SC 699
Course title	Research Project
Credits	30
Compulsory/optional	Compulsory
Prerequisites	GPA of 3.00 at M.Sc. (Course work)
Time allocation	3000 notional hours
Aims	The overall aim is to prepare the student to conduct a research independently. Specific aims: 1. To train students to apply scientific method in scientific research. 2. To train students to generate researchable hypotheses. 3. To train students to plan, design and conduct scientific research. 4. To gather reliable scientific data, analyse, and interpret. 5. To develop skills in scientific writing.
Intended learning outcomes	At the end of the successful completion of the course, students will be able to, 1. Apply the scientific method. 2. Design a research project. 3. Complete a research project. 4. Describe ethical issues in scientific research. 5. Explain the patenting process in research. 6. Make presentations at national/international conferences. 7. Produce a thesis conforming to the requirements of the PGIS. 8. Write manuscripts for publication in refereed journals.
Content	The students will conduct sufficient amount of work on a chosen research topic under the guidance provided by an assigned supervisor/s, make a presentation of research findings at a national/international conference, and produce a thesis.

Assessment criteria:

Continuous assessment	End-semester examination
30%	Oral examination (20%) Thesis (40%) Final presentation (10%)

Recommended Texts:

1. Backwell, J. and Martin, J. (2011) *A Scientific approach to Scientific writing*, Springer.
2. Postgraduate Institute of Science (2016) Guidelines for writing M.Sc. Project Report/M.Phil. Thesis/ PhD Thesis

Note: The format of the Thesis is available in the PGIS website.

CONTENTS OF OTHER COURSES

SC 411: Introduction to Computer Science *(no credits)*
Introduction and overview, Overview of the computer science curriculum, Intelligent machines and systems applications, Chemical, biological, and medical applications, Environmental and ecological applications, Information and educational applications, Engineering and scientific applications, Business and management applications, Communications and media applications. Introduction to Computer Programming: Basic concepts, Basic components of programming languages; Variables, declarations, binding, procedures, functions, Simple algorithms operating on nonstructured data, Modularity in computer programming; Basics of constructing larger programs: abstraction and instantiation of program components, Structured data; Lists, stacks, queues, ordered binary trees. Storing and accessing data structures, Operating on mutable data; Working with mutable data, object-based programming, data encapsulation. (Sections are covered using C, C++ and JAVA)

Recommended texts:

1. *Manis & Little, The Schematics of Computation, Prentice-Hall, 1995.*
2. *Laboratory work: Computer programming in C. Students are also required to write a weekly essay on the course material.*

SC 412: Introduction to theory of computation *(no credits)*
Sets, Propositions, Two-valued Boolean algebra, Inductions, Recursion, Relations and functions, Graphs, Basic Techniques; Mathematical proofs, induction and recursion, gcd, Fibonacci numbers, Lame's theorem, Counting; Rules of sum and product, permutations, combinations. Pascal's triangle, binomial theorem, summation of binomial coefficients, Probability; Probability, inclusion/exclusion, conditional probability, analysis of expected behavior of algorithms, Recurrences; Method of operators/annihilators, divide and conquer recurrences/algorithms, Basic Algorithmic Techniques; Dynamic programming, greedy heuristics, Graphs and Trees; Definitions, Hamiltonian paths and Ore's theorem, depth first search and applications, Eulerian paths, breadth first search and applications, planarity/Platonic solids, Automata and Languages; Finite state machines, regular languages/closure properties, pumping lemma, context free languages/pumping lemma/BNF, Turing machines/computability.

Recommended texts:

1. *Kenneth Rosen, Discrete Mathematics and Its Applications, 3rd edition, McGraw-Hill,*
2. *1990, Cormen, Leiserson, and Rivest, Introduction to Algorithms, McGraw-Hill.*

SC 413: Data structures and software principles *(no credits)*
Introduction, Program design concepts, Abstract data types, Basic data structures, Abstract data types for sets: operations and implementations, Sorting, Memory management, Graph algorithms, String algorithms, Arrays, records, pointers, indices, Recursion, Timing comparisons, Memory comparisons, Lists; implementation: array/linked; ordered/unordered, Searching: introduction to set abstract data type, Stacks and queues, Trees; Pointer implementation; traversal, Binary search trees; Definition, Searching, Creation and insertion, Good and bad trees, Deletion, B-trees, Hashing: initial hash, collisions, separate chaining, Graphs; Implementation, Depth first search, breadth first search, topological numbering, connected components, Sorting; Insertion sort, Quicksort, Heap as priority queue; heapsort.

Recommended texts:

1. *Mark Weiss, Efficient C Programming, Prentice-Hall, 1995. Mark Weiss, Data Structures*
2. *and Algorithms Analysis in C++, Benjamin Cummings. Watts Humphrey, Introduction to Personal Software Process, Addison Wesley.*
3. *Laboratory work: Computer programming in C and C ++.*

SC 414: Introduction to Computer Architecture *(no credits)*
Combinational logic networks, Computer arithmetic; arithmetic/logic unit, Sequential logic networks, Memory hierarchy, CPU design, I/O architecture, Instruction sets, addressing modes, linking and loading, Subroutines, ALU design, Basic processor design, Basic pipelining, Memory hierarchy design, Input/output, Parallel processing.

Recommended texts:

1. *Hennessy and Patterson, Computer Organization and Design: the Hardware/Software Interface, Morgan Kaufmann. M. M. Mano and C. R. Kime, Logic and Computer Design Fundamentals, 1997, Prentice Hall.*
2. *Laboratory work: Computer programming in assembler and electronics lab.*

SC 415: Programming and electronics laboratories *(no credits)*
Programming laboratory: Language constructs; Variables, assignments, loops, decision structures, input/output, files, subprograms/procedures, numeric and nonnumeric data. Design and construction of software; Top down and bottom up design, decomposition, structuring, design for reuse, documentation, study of examples, writing software as a team, using software from others. Programming assignments; A variety of progressively more complex assignments. Electronics laboratory: Review of basic features of computer hardware and software; Lab: Introduction to equipment, demo, simple experiment. Input/Output (I/O) concepts and examples; Lab: Experiment involving parallel I/O. More concepts and examples; Lab: Experiment involving serial I/O. Interfacing to the analog world; Lab: Experiment using digital-to-analog (D/A) conversion. Lab: Experiment using analog-to-digital (A/D) conversion. Techniques for analysis of acquired data; Lab: Experiment requiring digital signal processing. Interfacing to local area network (LAN); Lab: Experiment using LAN.

Recommended texts:

1. *Programming laboratory: C: Deitel and Deitel, C How to Program, Prentice-Hall. C++:*
2. *Deitel and Deitel, C++ How to Program, Prentice-Hall. Java: Deitel and Deitel, Java How to Program, Prentice-Hall.*
3. *Laboratory work: Computer programming and electronics labs.*

SC 416: Seminar *(no credits*)
Students are expected present a seminar on a selected topic approved by the programme coordinator

SC 417 - Data Structures and Algorithms *(no credits)*
Data Structures: linear and non linear data structures. arrays, lists: linked list, ordered linked list and doubly linked list; push down stacks; queues: FIFO queue and deque. Tree structures – trees in general, binary search three (BST), root insertion to BST, splay tree, 2-3-4 trees, radix tree and red-black tree; Graphs; Implementation of depth first search, breadth first search; Analysis of algorithms: time complexity, big O notation. Sorting algorithms: bubble sort, selection sort, insertion sort, quick sort, heap sort, merge sort and external sorting methods. Hashing: hash functions and collision resolution: separate chaining, linear probing and double hashing. Classification of Algorithms by Implementation and Design Paradigm: Divide & Conquer Algorithms, Dynamic Programming, Greedy Algorithms, Recursive Algorithms, Backtracking, Alfa-Beta pruning, Branch & Bound Search;

Recommended texts:

1. *Sara Baase, Allen Van Gelder, Computer Algorithms - Introduction to Design & Analysis, Addison-Wesley, 2000*
2. *Thomas H. Cormen, Charles E. Leiserson & Ronald L. Rivest Introduction to Algorithms, McGraw-Hill, 200,*
3. *Mark Weiss, Efficient C Programming, Prentice-Hall, 1995.*

SC 418 - Programming techniques *(no credits)*

Basic Concepts: The structure & definition of a HLL such as C, the concept of Data types and operation on data types. Structured Programme Development: Problem definition and specification, top-down design and development, Coding guidelines & standards in developing commercial application systems. Writing a complete program: Sequential, alternation, and repetition control structure: formatted and unformatted basic input output, Modular structure programme modules in C, functions. Pointers: Pointers concept, operations on pointers and usage of pointers. Array processing Character and string processing. Simple sorting and searching algorithms Bubble sort, sequential and binary search. File processing: File Definition; processing logic for sequential and random files. Classification of Data types and Data Structure, scalar and structured data types, static and

dynamic structures. Testing of programme via both black box and white box testing techniques and system integration via bottom -up or top-down approach.

Recommended texts:

1. R. Sedgwick, Algorithms in C, Addition Wesley, 1998.
2. Standish, T. A, Data Structures in Java, Addition Wesley, 1998.
3. Gregory L. Heilemen, Data structures, Algorithms & Object-Oriented programming, McGraw-Hill, 1996.
4. Sara Baase, Allen Van Gelder, Computer Algorithms - Introduction to Design & Analysis, Addison-Wesley, 2000

SC 531: Database systems *(2 credits*)

Introduction; An overview of a database management system, The entity-relationship model, Logical organization of databases; The relational model, Relational algebra, SQL, Examples of existing relational database management systems, Physical organization of databases; Characteristics of disks and disk storage, Storage of relations, Indexing: B-trees and hashing, Query processing and optimization, Concurrency control; Transaction, Serializability, Locking, Logging and recovery, Distributed databases, Functional dependencies and normal forms, Information services for unstructured data.

Recommended texts:

1. *Elmasri and Navathe, Fundamentals of Database Systems, 2nd edition,*
2. *Benjamin/Cummings.*
3. *Laboratory work: Computer programming using database management packages such as Informix,*
4. *Oracle and FoxPro on PCs and UNIX workstations.*

SC 532: Combinatorial mathematics *(2 credits)*

Introduction, Numbers and counting, Subsets, partitions, permutations, Recurrence relations and generating functions, The principle of inclusion and exclusion, Latin squares and SDRs, Extremal set theory, Steiner triple systems, Finite geometry, Ramsey's theorem, Graphs, Posets, lattices, and matroids, Automorphism groups and permutations, Enumeration under group action, Designs, Errorcorrecting codes.

Recommended texts:

1. *Peter Cameron, Combinatorics.*

SC 533: Introduction to parallel computing *(2 credits*)

Ideal and real machine models; Vector (pipelined) processors, array machines, shared-memory multiprocessors, message-passing multiprocessors, others; programming constructs native to each class of machine. Programming models and their languages; Data-parallel models (array parallelism, parallel loops), process-based models; illustrative examples, such as matrix multiplication, sorting, and the n-body problem. Cost models and efficiency analysis of parallel programs, Parallel programming issues; Locality, grain size, scheduling, load balancing, data distribution and alignment, communication analysis, synchronous programming, determinacy and nondeterminacy. Debugging parallel programs, Performance measurement, evaluation, and tuning, Case studies; Simple case studies from application areas such as computational fluid dynamics, computational biology, and operations research. Discrete event simulations, Grid-structured computations, Tree-structured computations, Sparse and dense linear systems, Parallelizing, Compilers.

Recommended texts:

2. *Almassi-Gottlieb, Highly Parallel Computing, 2nd edition, Benjamin Cummings, 1994.*
3. *Michael Quinn, Parallel Computing: Theory and Practice, 2nd edition, McGraw Hill, 1994. Kumar,*
4. *Grama, Gupta, and Karypis, Introduction to Parallel Computing, Benjamin Cummings, 1994. Ian*
5. *Foster, Design and Building of Parallel Programs, Addison Wesley.*
6. *Laboratory work: Implementation of parallel algorithms on one or more classes of parallel computers. Emphasis is on numerical algorithms.*

SC 534: Programming language design and compilers *(2 credits)*
Language Design: Elements of imperative languages, Data types: arrays, lists, user-defined types, Functional programming, Control operations, Object-oriented programming, Types Compilation: Lexical analysis: transition diagrams, regular expressions, using lex, Syntactic analysis: context-free grammars, top-down and bottom-up parsing, using yacc, Abstract syntax; syntax-directed translation, Code generation.

Recommended texts:

1. *Ravi Sethi, Programming Languages, 2nd edition, Addison-Wesley. Jim Holmes, Building*
2. *Your Own Compiler With C++, Prentice Hall.*
3. *Laboratory work: Computer programming on workstations.*

SC 535: Operating system design ***(****2 credits)*
Processes and concurrent programming; Basic concepts: states, transitions. Mutual exclusion, synchronization, semaphores, monitors, Ada rendezvous. Deadlock and indefinite postponement; prevention, avoidance, detection, recovery. Operating system components; Real and virtual memory; paging and segmentation; fetch, placement, and replacement algorithms; thrashing. Processor scheduling; disk space management and allocation; seek and rotational optimization; blocking and buffering. File systems; directory, structures; access methods; access control. Advanced topics; Performance evaluation. Distributed and parallel operating systems. Object orientation. Security and protection; encryption. Case Studies.

Recommended texts:

1. *Either H. M. Deitel, Operating Systems, Prentice Hall; or Silberschatz and Gavin,*
2. *Operating System Concepts, 5th edition, Addison Wesley.*
3. *Laboratory work: Computer programming on UNIX workstations.*

SC 536: Graph Theory *(2 credits)*
Introduction, Graphs: Graphs and simple graphs; Graphs isomorphism; The incidence and adjacency matrices; Vertex degrees; Paths and connection; Cycles and the shortest path problem, Trees: Trees; Cut edges and bonds; Cut vertices; Cayley's formula and Kruskal's algorithm, Connectivity: Connectivity; Blocks and construction of reliable communication networks, Euler Tours and Hamilton Cycles: Euler tours; Hamilton cycles; The Chinese postman problem and the travelling salesman problem, Planar Graphs: Planar graphs; Dual graphs and Euler's formula, Networks: Flows; Cuts; The Max-Flow Min-Cut theorem and applications.

SC 537: Computer Networks & Distributed systems *(2 credits)*
Introduction; Examples of computer networks and distributed systems, Concept of layered architecture. ISO reference model of Open System Interconnection. Overview of communication subnetworks; Physical layer protocol issues, Data link layer protocols, Network layer protocol issues: Virtual circuits vs. datagrams. Local-area network architectures, Satellite and packet radio networks. Point-to-point packet switched networks. Models of network interconnection. Standard network access protocols. Transport and session protocol design issues; Transport connection and connection establishment, Flow control and buffering. Synchronization in distributed environment. Multiplexing. Crash recovery. Networking facilities in well-known systems. Presentation layer protocols; Terminal handling and protocols. File transfer protocol design issues. Network security and privacy. Standards for presentation layer protocols. Distributed operating system design; Models and primitives of distributed computing, Distributed resource management and scheduling: File allocation. Load sharing. Task assignment, etc. Distributed database systems; Concurrency control and synchronization. Current topics. Examples of distributed database systems.

Recommended texts:

1. *Sape Mullender, Distributed Systems, 2nd edition, Addison-Wesley.*

SC 538: Artificial Intelligence (*2 credits*)
Introduction; Organization and overview, Program inspection: CHAT, a question answering program, Introduction to LISP, Program inspection: Robert's computer vision program, LISP II, Knowledge representation and deduction; Knowledge representation and valid arguments, Propositional Predicate Calculus (PPC): syntax and semantics, Deduction in PPC: rule-based systems & search spaces, Deduction in PPC: conjunctive goals & answer extraction, First-Order Predicate Calculus (FOPC): syntax and semantics, Representing knowledge in FOPC, Unification, Deduction in FOPC. Problem solving and search; Game trees and search, Basic search techniques, Problem solving: partial instantiate and prune, Generation of search heuristics. Planning; Introduction to planning. Non-linearity and protection intervals. Plan representation. Natural language processing; Introduction to natural language, Case grammar, Conceptual analysis, ATNs and review. Computer vision (understanding line drawings); Guzman's system & Huffman labeling, Waltz labeling and constraint propagation.

Recommended texts:

1. *Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice-*
2. *Hall, 1995. Patrick Winston and B. K. Horn, LISP, 3rd edition, Addison Wesley.*

SC 539: Advanced topics in computer graphics *(2 credits)*
Review of computer graphics fundamentals, Nonparametric object representations; Conics, algebraic surfaces, bump functions. Parametric object representations; Quadrics, superquadrics, splines. NonEuclidean representations; Fractals, particle systems. Rendering; Lighting models, fast-Phong algorithm, A-buffer, V-buffer, radiosity. Ray-tracing algorithms; Distributed methods, space subdivision, parallel methods. Texture mapping, Animation; Key-frame systems, animation languages, kinetic vs. dynamic systems, modeling human and animal motion. Scientific data visualization.

Recommended texts:

1. *Hearn and Baker, Computer Graphics - C Version, 2nd edition, Prentice Hall.*
2. *Laboratory work: Computer programming on UNIX workstations.*

SC 540: Web Servers and Technologies (2 credits)
Introduction to Client Side Scripting, Java Script: JavaScript syntax, JavaScript object model, JavaScript objects, Static objects, Forms objects, Event handling - Mouse related events, Keyboard events, Document events, Output in JavaScript, Introduction to VB Script. ASP.net: Implement ASP.net with VBScript, Use SQL & ADO to Interact with ASP.net Databases, Write Cookies on the Client Using ASP.net, J2EE - Java Enterprise Edition: JDBC, JSP, Servlets, Hypertext Preprocessor: Program structure, Use php to process html forms, Regular expressions for form validation and other applications, Read and write files, Database applications. XML: Understand the role of XML, Write XSL Documents to Describe how XML Documents are to HTML, Create Simple DTD & Schema Files to Describe the Grammar of XML, Differences between DTD's & Schema, Differences between Cascading Style Sheets & XSL, Other new trends in Web development: Eg. SOAP, WSDL

Recommended texts:

1. *Java How to Program: Java 2, Introducing Swing, Harvey M. Deitel , Harvey M. Deitel, Paul J. Deitel , Paul J. Deitel, Prentice Hall Professional Technical Reference, 1999*
2. *Benoit Marchal, XML by Example ,2nd Edition, Que Publisher Internet, 2001*
3. *World Wide Web How to Program Second Edition, 2002*

SC 541: Programming Web Applications
Explores the use of scripting languages, such as Java Script, PHP, and Java Applets in web site development. Examines the use of relational databases to create dynamic web sites. Extensive exposure in lecture and lab to web based application development tools. Students will develop a full-featured web based interactive educational application.

Recommended texts:

1. *Benoit Marchal, Internet & World Wide Web How to Programme, Second Edition, , 2002*
2. *Java How to Program: Java 2, Introducing Swing, Harvey M. Deitel , Harvey M. Deitel, Paul J. Deitel , Paul J. Deitel, Prentice Hall Professional Technical Reference, 1999*
3. *Benoit Marchal, XML by Example ,2nd Edition, Que Publisher Internet, 2001*

SC 542: Introduction to Wireless and Mobile Systems
Overview of Wireless Networks; Fundamentals of Cellular Networks; Cellular Networking (1G, 2G); Next Generation Cellular Networks (GPRS, UMTS, 3G and Beyond); Wireless Local Area Networks (WLANs); Wireless Personal Area Networks (WPANs); Mobile IP; Ad Hoc Networking; Wireless Sensor Networks; Wireless Mesh Networks

Recommended text:

1. *Dharma P. Agrawal, Introduction to Wireless and Mobile Systems, University of Cincinnati, 2011.*
2. *T. S. Rappaport, Wireless Communications: Principles and Practice, Prentice Hall, 2002*

SC 543: Object Oriented Analysis and Design
Fundamental of Object-oriented design: Encapsulation, classes and objects, information hiding, operator overloading, inheritance, overriding, delegation; Analyze problems, determine objects that are necessary to model the system, determine what attributes the objects need to have, determine what behaviors the objects need to exhibit, develop conceptual models, generate designs from the models.

Recommended text:

1. *Grady Booch, Object-Oriented Analysis and Design with Applications, 2nd Edition, Addison-Wesley Professional, 1993*
2. *Michael J. Quillin, Object Oriented Analysis and Design, 2005*
3. *An Introduction to Object-Oriented Analysis and Design and the Unified Process Second Edition. Craig Larman, 2002.*

SC 546: Software engineering *(2 credits*)
The software life cycle: cost of software and hardware, Software quality, User requirements, System specifications; Survey of abstract data types, axiomatic and operational techniques, concurrency. Design fundamentals; Abstract machines, stepwise refinement, simulation, bottom-up approach, modularity, extensions from a nucleus, techniques for real-time systems. Design techniques, Programming; Language requirements for software engineering, language specifications, high-level/low-level/assembler tradeoffs, concurrency, real-time programming, team programming, optimization. Testing: theoretical and empirical; cost of testing, Verification: Partial correctness, proving termination, Maintenance; Portability, adaptability, modification, distribution. Reliability; Redundancy, error detection, fault-tolerance, faults, failures, recovery. Protection and security, Management of software projects: manpower, case studies.

Recommended texts:

1. *Roger Pressman, Software Engineering: A Practitioner's Approach, 4th edition,*
2. *Laboratory work: Computer programming on workstations*.

SC 547: Computer architecture (*2 credits*)
Technology and performance, Instruction set architectures, Computer arithmetic, Central processing units, Pipelining, Memory hierarchies, Input-output mechanisms, Vector and multiprocessors, Parallel programming features, Case studies.

Recommended texts:

1. *Patterson and Hennessy, Computer Organization and Design: the Hardware/Software*
2. *Interfaces, Morgan Kaufmann, 1994. Stone, High-Performance Computer Architecture, 3rd edition,Addison Wesley, 1993.*

SC 548: Systems analysis/Systems engineering (*2 credits*)
Introduction: Systematic Thinking, Systems Definitions, Classification of Systems, Computer Based Systems. Systems Analysis: Models of Systems, Formal Methods UML, Logical Algebra, Automata, Simulations. System Engineering: System Life Cycle Processes, System Life Cycles Stages, CASE, Sociological Systems, Soft System Methodology, System Philosophy, Review & Trends.

SC 549: Artificial Neural Networks (*2 credits*)
Elementary neurophysiological principles, Artificial neuron models, Single layer networks (perceptions), Multi-layer feed forward networks (+back propagation), Cascade correlation (correlation training), Recurrent networks (Hopfield), Self-organizing maps (Kohonen maps), Bidirectional associative memory, Counter propagation networks, Adaptive resonance theory, Spatiotemporal sequences, Hardware realization of neural networks, Individual projects.

Recommended texts:

1. *Simon Haykin, Neural networks: A Comprehensive Foundation, Macmillan, 1994.*
2. *Laboratory work: Computer programming on UNIX workstations.*

SC 550: Linear programming (*2 credits*)
Review, Geometry in IRn, Introduction to linear programming, Further topics, Integer programming, Special types of linear programming problems.

Recommended texts:

1. *Kolman and Beck, Elementary Linear Programming with Applications, Academic Press, 1980.*

SC 551: Communication networks for computers *(2 credits)*
Overview; Examples and concepts of layered architecture; overview of higher layer protocols. Transport Layer; Internet addressing and Internet protocols; socket interface; TCP/IP protocols; client-server models. Network layer; Taxonomies; relevant parameters of network and traffic. Performance evaluation and queuing theory. Multiple-access methods for broadcast networks; Taxonomies of multiple access methods; contention methods; polling methods; reservation methods. Switched networks; Architectures of switches: circuit, packet, and ATM switches; scheduling and admission control; routing, flow control, and congestion control. Interconnections of networks, Logical data link protocols.

Recommended texts:

1. *A. Tanenbaum, Computer Networks, 3rd Edition. D. Comer, Internetworking with TCP/IP, 2nd Edition, Volume 1.*
2. *Fred Halsall, Data Communications, Computer Networks, and Open Systems,3rd edition, Addison-Wesley.*

SC 552: Digital image processing *(2 credits)*
Introduction to image processing, Elements of a digital image processing system; image acquisition, storage, processing, transmission and display. Image processing fundamentals; human vision system, sampling and quantization (spatial and brightness resolution), pixels and their relationships. Digital image processing techniques; image enhancement and restoration, pixel point processing, pixel group processing, frequency domain processing (Fourier transform), geometric transformations, image analysis, segmentation, feature extraction. Image compression and transmission, run-length encoding. Coding systems; error detection and correction, data compression schemes. Pattern recognition; basic concepts, clusters, decision functions, cluster seeking algorithms.

Recommended texts:

1. *Digital Image Processing, Gregory A. Baxes, SR 621.367*
2. *Other reading: Digital Image Processing; Remote Sensing and Image Interpretation, T. M. Lillesand and R. W. Kiefer; Remote Sensing Digital Image Analysis, J. A. Richards*

SC 553: Project management *(2 credits)*
Principles of Project Planning, Project Initialisation, Project Life Cycles & Planning, Identifying Tasks and Estimating, Product Planning, Quality Issues, Anticipating Problems & Motivation, Financial Issues, Applying Principles.

SC 554: Special topics in Computer Science *(2 credits)*
Lecture course in topics of current interest.

Recommended texts:

1. *Depends on the topic.*
2. *Laboratory work: Depends on the topic.*

SC 555: Laboratory work *(2 credits)*

Students of the batch are organized into teams of four to six students with an academic advisor to analyze a problem proposed, to select a suitable solution, and to implement that solution. Students work in teams to solve typical commercial or industrial problems. Work involves planning, design, and implementation. Oral and written work is required.

SC 556: Computer Vision

Discrete geometry and quantization, length estimations, automated visual inspection, object recognition and matching, depth perception problems, stereo geometry and correspondence, motion analysis, optical flow, applications of Computer Vision, remote sensing, biomedical imaging, document processing, target tracking

Recommended text:

1. *Computer Imaging: Digital Image Analysis and Processing* , SE Umbaugh, CRC Press, 2005
2. *Computer Vision: A Modern Approach by David A. Forsyth and Jean Ponce, Prentice Hall, 2002*

SC 557: Analysis of Communication Networks

Mathematical Foundations: Probability, Random Processes, Markov Chains; Queuing Analysis: Queue performance, Performance bounds, Systems of communicating markov chains; Modeling traffic flow control protocols: Leaky bucket algorithm, Token bucket algorithm, Virtual scheduling algorithm; Modeling error control protocols: Stop-and-wait ARQ, Go-back-n ARQ, Selective Repeat ARQ; Modeling access control protocols: Aloha, Slotted Aloha, IEEE 802.3 (CSMA/CD), CSMA/CA

Recommended text:

1. George Kesidis, An Introduction to Communication Network Analysis, Wiley-IEEE Press; 1st edition , 2007
2. Fayez Gebali, Analysis of Computer and Communication Networks, Springer-Verlag, 2008

SC 558: Cryptography and Network Security

Introduction to the subject and related mathematics; Traditional cryptography
Modern cryptography – symmetric and asymmetric systems; Cryptographic applications in data networks; Design of a cryptographically protected system

Recommended text:

1. *Behrouz Forrouzan, Cryptography and Network Security,* McGraw-Hill, 2007

SC 559: Multimedia Systems

Introduction and overview: Discrete Cosine Transform Coefficient Coding. *Audio Coding:* Analogue and digital form: Sample rate, bits/sample, nyquist rate, CD audio Compression techniques: - PCM, ADPCM, LPC, GSM/CELP, MP3/AAC. *Video:* TV Standards: Interlacing vs progressive scan, PAL, NTSC, SECAM Video digitization, Raw Image Representation: RGB, YUV411, YUV422, Indexed color vs true color Image Compression: - GIF, JPEG, Motion JPEG: Video Compression: - Motion estimation - Motion compensation Video Compression Schemes: - H.261, H.263 - MPEG 1, MPEG 2, MPEG 4 Video Adaptation: - Sender-side adaptation, buffering, VBR->CBR conversion. *System Streams:* MPEG program and transport streams H.221 framing (for ISDN) IP-based transport: - packet loss - TCP vs. UDP - Application-level framing – RTP - H.261 as example of payload format – DCCP Audio/Video synchronization – RTCP - MPEG system stream. *Signaling:* H.323 SIP and SDP RTSP Megaco. *OS Issues:* Buffering Scheduling. *Describing Network Traffic:* Traffic patterns Application requirements QoS parameters and descritions. *Congestion control and Resource Management:* TCP congestion control Real-time traffic congestion control Queue management: - Random Early Detection + other AQM - Explicit Congection Notification (ECN) - Scheduling mechanisms (FQ, WFQ). *Enhanced Quality of Service:* Intserv Resource reSerVation Protocol (RSVP) Diffserv. *IP Multicast:* Service Model Layered transmission Multicast congestion control. *Digital rights management:* Legal issues Watermarking

Recommended Text:

1. *Introduction to Multimedia Systems (Communications, Networking and Multimedia), by Sugata Mitra and Gaurav Bhatnagar*

SC 560: Multimedia practical
Introduction to multimedia packages, sound editing, video editing, 2D and 3D animation design

Recommended Text:
1. *Introduction to Multimedia Systems (Communications, Networking and Multimedia), by Sugata Mitra and Gaurav Bhatnagar*

SC 561: Data mining Techniques
Introduction: Basic Data Mining Tasks, Database / OLTP Systems, Data Warehousing, OLAP Systems, Related Concepts (Statistics, Fuzzy Sets and Fuzzy Logic, Information Retrieval, Decision Support Systems, Dimensional Modeling, Machine Learning, Pattern Matching). Data Preprocessing, Exploratory Data Analysis, Statistical Approaches to Estimation and Prediction. Association Rule Mining. Classification and Prediction: Introduction, Decision Tree Induction Methods, Bayesian Classification, Rule Based Algorithms, Neural Network Based Algorithms. Cluster Analysis: Introduction, Similarity and Distance Measures, Partitioning Methods, Hierarchical Methods, Outlier Analysis. Web Mining: Web Content Mining, Web Structure Mining, Web Usage Mining. Applications and Trends in Data Mining. Some practical assignments will be given for this course.

Recommended Texts:
1. *Data Mining Introductory and Advanced topics,* M.H. Dunham, 2003
2. *Predictive Data Mining,* Weiss SM & Indurkhya N, Morgan Kaufmann, 1997
3. *Principles of Data Mining,* Hand DJ et al, MIT Press, 2001

7. PROGRAMME EVALUATION

Evaluation of Course work

Based on the scheme given below, the overall performance of a student in a given course shall be evaluated by the respective instructor(s) and a grade shall be assigned.

Evaluation Scheme
- For all courses a minimum of 80% attendance is expected.
- The evaluation of each course shall be based on within course and end of course examinations, and assignments. The weightage of marks given below can generally be used as a guideline in the computation of the final grade.
  End of course examination 50 - 60%
  Continuous assessments (mid-semester examination, assignments, etc.) 40 - 50%
- Courses with laboratory and/or fieldwork shall be evaluated, where applicable, on a continuous assessment basis.
- The minimum grade a student should achieve to pass a course is C.
- Students will be informed of the evaluation scheme by the instructor at the beginning of a given course.

Grade Points and Grade Point Average (GPA)
The Grade Point Average (GPA) will be computed using the grades earned for core courses and optional courses, taken for credit. Preliminary courses, industrial training, research project and seminar will be evaluated on a pass/fail basis.

On completion of the end of course examination, the instructor(s) is/are required to hand over the grades of a given course to the programme coordinator who will assign the Grade Points using the following table:

Grade Grade Point

A+	4.0
A	4.0
A^-	3.7
B^+	3.3
B	3.0
B^-	2.7
C^+	2.3
C	2.0
F	0.0

The Grade Point Average (GPA) will be computed using the formula:

$$GPA = \frac{\Sigma\, c_i g_i}{\Sigma\, c_i},$$ where c_i = number of credit units for the i^{th} course, and g_i = grade point for the i^{th} course

Make-up Examinations

'Make-up' examinations may be given only to students who fail to sit a particular examination due to medical or other valid reasons acceptable to the PGIS.

Repeat Courses

If a student fails a course or wishes to improve his/her previous grade in a course, he/she shall repeat the course and course examinations at the next available opportunity. However, he/she may be exempted from repeating the course, and repeat only the course examinations if recommended by the teacher-in-charge or M.Sc. Programme Coordinator. The student may repeat the same course or a substituted (new) optional course in place of the original course. A student is allowed to repeat five credits of coursework free-of-charge. The maximum number of credits a candidate is allowed to repeat is fifteen. The maximum grade, a candidate could obtain at a repeat attempt is a B and he/she is allowed to repeat a given course only on two subsequent occasions.

Evaluation of Research Project

Research project will be evaluated on the basis of a written report (M.Sc. project report) and oral presentation (see Section 6.0 of the PGIS Handbook for the format of the project report).

8. PANEL OF TEACHERS

Name, affiliation/Address, and qualifications	Area of Specialization
Dr. H.R.O. E. Dayaratne Dept. of Statistics and Computer Science, Faculty of Science, Univ. of Peradeniya B.Sc. (Perad.), Ph.D. (Keio, Japan)	Mobile Networks, Internet of Things, Mobile Cloud Computing, Software Defined Networks (SDN), Mobility protocols
Dr. D. Elkaduwa Dept. of Computer Engineering, Faculty of Engineering, Univ. of Peradeniya B.Sc.Eng (Perad.), Ph.D. (UNSW, Australia)	Operating Systems, Formal Verification, GPU Programming
Mr. P.M.P.C. Gunathilake Dept. of Statistics and Computer Science, Faculty of	Internet of Things, Machine Learning, High Performance Computing, Image

Science, Univ. of Peradeniya M.Phil. (Perad.)	Processing
Prof. S. R. Kodituwakku Dept. of Statistics and Computer Science, Faculty of Science, Univ. of Peradeniya B.Sc. (Perad.), M.Sc. (AIT, Thailand), Ph.D. (RMIT, Australia)	Software Engineering, Database Management, Distributed Computing, Image Processing and Retrieval
Prof. K. M. Liyanage Dept. of Electrical and Electronic Engineering, Faculty of Engineering, Univ. of Peradeniya B.Sc. Eng. (Perad.), M.Eng., D. Eng. (Tokyo, Japan)	Energy, Sustainability Science, Communication Networks, Information Security, Optimization and Control
Dr. R. D. Nawarathna Dept. Statistics and Computer Science, Faculty of Science, Univ. of Peradeniya B.Sc.(Perad.), M.Sc., Ph.D. (North Texas, USA)	Intelligent Systems, Machine Learning, Deep Learning, Computer Vision, Medical Image Processing
Dr. I.B. Nawinne Dept. of Computer Engineering, Faculty of Engineering, Univ. of Peradeniya B.Sc.Eng (Perad.), Ph.D. (UNSW, Australia)	Computer Engineering Embedded Systems
Prof. A. A. I. Perera Dept. of Mathematics, Faculty of Science, Univ. of Peradeniya B.Sc. (Perad.), M.Sc. (Oslo), Ph.D. (RMIT, Australia)	Graph Theory, Algebra Discrete Mathematics, Design Theory
Dr. U. A. J. Pinidiyaarachchci Dept. of Statistics and Computer Science, Faculty of Science, Univ. of Peradeniya B.Sc. (Perad.), Ph.D. (Uppsala, Sweden)	Digital Image Processing, Computer Vision, Biomedical Engineering, Cell Image Analysis, Pattern Recognition, Data Science
Dr. R. Ragel Dept. of Computer Engineering, Faculty of Engineering, Univ. of Peradeniya B.Sc.Eng (Perad.), Ph.D. (UNSW, Australia)	Microarchitectural Aspects of Embedded Systems Design, High Performance Computing, Application Specific Processor Design
Dr. M. Sandirigama Dept. of Computer Engineering, Faculty of Engineering, Univ. of Peradeniya B.Sc. Eng (Perad.), M.Sc., Ph.D. (Ehime, Japan)	Secure Communication Protocols, and Legal Aspects of Technology
Dr. R. Siyambalapitiya Dept. of Statistics and Computer Science, Faculty of Science, Univ. of Peradeniya B.Sc. (Perad.), Ph.D. (Perad.)	Design of Algorithms, Optimization Scheduling, Operating Systems
Prof. R. D. Yapa Dept. of Statistics and Computer Science, Faculty of Science, Univ. of Peradeniya B.Sc. (J',Pura)), M.Sc. (Col.), Ph.D.(Hiroshima, Japan)	Crime Data Analysis, Data Science, Big Data Analytics, Spatial Analysis, Social Network Analysis, Medical Image Processing

9. PROGRAMME COORDINATORS

Prof. Athula Perera
Department of Mathematics
Faculty of Science
University of Peradeniya
Peradeniya
Tel: 081-2394560
Email: athula103@yahoo.com

Dr. Ruwan Nawarathna
Department of Statistics and Computer Science
Faculty of Science
University of Peradeniya
Peradeniya
Tel: 076 755 2263
Email: ruwand@pdn.ac.lk