Department of Computer Science

FACULTY OF SCIENCE
UNIVERSITY OF JAFFNA
SRI LANKA

Effective from the Academic Year 2019/2020

Curriculum for
Bachelor of Science Honours in Computer Science
Level 4S

December 2020

Detailed Syllabi

Bachelor of Science Honours in Computer Science

Level 4S

Course Code:	CSC401S3		
Course Title:	Advanced Algorithms		
Credit Value:	03		
Hourly Breakdown:	Theory	Practical	Independent Learning
	45	--	105

Objectives:

Provide in-depth knowledge for designing efficient algorithms using appropriate data structures and a variety of advanced computational techniques.

Intended Learning Outcomes:

- Perform algorithm analysis using appropriate techniques
- Apply algorithms for solving problems of various complexity
- Identify suitable data structure(s) for efficient problem solving
- Formulate a real world problem into a model solvable by standard algorithmic approaches
- Defend NP-Hard & NP-Complete problems

Course Contents:

- Analysis of algorithms: Order of growth and asymptotic notation, master method, amortised analysis
- Algorithms for sorting and order statistics: Standard sorting algorithms, sorting in linear time, medians, order statistics
- Advanced data structures: Hash table, binary heap, binary search tree, red-black tree, augmenting data structures, B-trees, Fibonacci heaps, utilising data structures for problem solving.
- Dynamic programming: Elements of dynamic programming (optimal substructure, overlapping sub-problems, memoization), application of dynamic programming approach for problem solving
- Greedy algorithms: Elements of Greedy strategy (optimal substructure, overlapping sub-problems, Greedy strategy), comparison of dynamic programming and Greedy approach, application of Greedy approach for problem solving
- P, NP, NP-Hard and NP-Completeness: Introduction to class of problems, NP-completeness (Cook's theorem), classic NP-complete problems, reduction techniques

Teaching and Learning Methods:

Lectures, Tutorial discussion, e-based teaching, Open Educational Resources, Assignments, Guided learning

Assessment Strategy

• In-Course Assessments	30%
• End-of-Course Examination	70%

References:

- Tardos, J. K. E., Algorithm Design, 2nd Ed., Pearson Education, 2011.
- Sedgewick, R., Algorithms, 4th Ed., Addison-Wesley Professional, 2011.
- Thomas, H. Cormen, T. Leiserson, C. E., Rivest, R. L., and Stein, C., Introduction to Algorithms, 3rd Ed., MIT Press, 2009.

Course Code:	CSC402S3		
Course Title:	Compiler Design		
Credit Value:	03		
Hourly Breakdown:	**Theory**	**Practical**	**Independent Learning**
	45	**--**	**105**

Objectives:

Provide in-depth knowledge of compiler components and principles involved in compiler design.

Intended Learning Outcomes:

- Discuss the functionalities of a compiler
- Represent a specified language using nondeterministic and deterministic finite automata
- Analyse the syntax of a language using top-down and bottom-up parsing approaches
- Discuss implementation techniques for a compiler
- Design algorithmic routines for type checking and code generation

Course Contents:

- Introduction: Compilers, Interpreters, and phases of compilation process
- Lexical Analysis: Regular expressions and their properties, converting regular expressions to Nondeterministic Finite Automata (NFA), Converting NFA to Deterministic finite automata (DFA), Eliminating dead states and minimising DFA
- Syntax Analysis: Context-free grammars, Syntax trees, Ambiguity in Grammar, Operator precedence, eliminating left-recursion, Left-factorisation, Top-down parsing methods, Bottom-up parsing methods, Resolving conflicts in parsing tables
- Scope and Symbol Tables: Dealing with scopes of identifiers, Efficient implementations of symbol tables
- Type Checking: The design space of types, Attributes, Type checking procedures
- Code Generation: Syntax Directed Translations, Intermediate-Code Generation, Machine-Code Generation

Teaching and Learning Methods:

Lectures, Tutorial discussion, e-based teaching-learning, Open Educational Resources, Assignments, Guided Learning

Assessment Strategy

• In-Course Assessments	30%
• End-of-Course Examination	70%

References:

- Mogensen, T. E., Introduction to Compiler Design, 2nd Ed., Springer, 2017.
- Lam, M. S., Sethi, R., and Aho, A. V., Compilers: Principles, Techniques, and Tools, 2nd Ed., Pearson Education, 2013.
- Galles, D., Modern Compiler Design, 2nd Ed., Pearson Education, 2009.

Course Code:	CSC404S3		
Course Title:	Information Systems Security		
Credit Value:	03		
Hourly Breakdown:	**Theory**	**Practical**	**Independent Learning**
	45	--	**105**

Objectives:

Provide knowledge to identify various security threats and propose suitable approaches to protecting Information Systems.

Intended Learning Outcomes:

- Identify various security threats and attacks on Information Systems.
- Explain security design principles.
- Elaborate techniques for data protection on Information Systems.
- Understand the techniques used to protect Computer Networks
- Explain the protocols that are utilised to protect the network and application layers of the Internet.

Course Contents:

- Introduction: Key security concepts, Critical characteristics of Information System, Secure-System life cycle, Security Professionals and the Organization.
- The need for security: Threats, Attacks, Secure software development.
- Security technologies: Firewalls, Virtual private networks, Intrusion detection and prevention systems, other security tools.
- Cryptography: Classical encryption techniques, Block ciphers, Data encryption standard, Advanced encryption standard, Public-Key cryptosystems.
- Cryptographic data integrity algorithms: Cryptographic hash functions, Message authentication codes, Digital signatures.
- Network security: Network access control and cloud security, Transport-level security, Wireless network security.
- Internet security: Internet Protocol security, web security (S-HTTP), email threats and email security (S/MIME)

Teaching and Learning Methods:

Lectures, e-based teaching-learning, Tutorial discussion, Assignments, Simulations, Use of Open Educational Resources, Guided Learning

Assessment Strategy

• In-Course Assessments	30%
• End-of-Course Examination	70%

References:

- Whitman, M. E. and Mattord, H. J., “Principles of Information Security”, 6th Ed., Cengage Learning, 2017.
- Stallings, W., “Cryptography and Network Security: Principles and Practice”, 4th Ed, Pearson Education Limited, 2017
- Paar, C. and Pelzl, J., “Understanding Cryptography: A Textbook for Students and Practitioners” 1st Ed, Springer, 2014.

Course Code:	CSC405S3		
Course Title:	Systems and Network Administration		
Credit Value:	03		
Hourly Breakdown:	Theory	Practical	Independent Learning
	15	60	75

Objectives:

Provide theoretical and practical knowledge required to manage and maintain hosts, network connectivity devices, and various networked servers.

Intended Learning Outcomes:

- Administer computer systems and connectivity devices
- Configure systems and devices for different networking scenarios
- Design a data communication network required for a small/medium organisation
- Demonstrate ability in network and server management
- Implement security policies in networked systems

Course Contents:

- Host Management: Host hardware and maintenance, Basic commands, Files, Directories and File System, Editors, Processes, Users and group management, Package management, Automating system administration.
- Network Design and Management: Network connectivity devices, Host network configuration, Routing and Router configuration, IP addressing, subnetting, Switch configuration, Wireless equipment, VLAN, Inter-VLAN routing,
- Server Management: Install and manage server operating systems, Web server, e-mail server, Proxy server, DNS server, Content Servers,
- Security Policy Implementation: Firewall configuration, IP tables, Secure remote administration, Simple Intrusion Detection Techniques, Snort.

Teaching and Learning Methods:

Lectures, Practical, e-based teaching-learning, Open Educational Resources, Assignments, Online based training, Simulation, Guided Learning

Assessment Strategy

• In-Course Assessment (Theory)	20%
• Group Project	20%
• End-of-Course Examination (Practical)	60%

References:

- Blokdyk, G., “Computer Network Administration: A Clear and Concise Reference”, 1st Ed, 5 Star Cooks, 2019
- Frisc, A., “Essential System Administration”, 3rd Ed, O'Reilly Media Inc., 2002.
- Odom, W., CCNA 200-301 Official Cert Guide Library, 1st Ed, Cisco Press, 2019
- Bauer, M. D., “Linux Server Security”, 2nd Edition, O'Reilly Media Inc., 2005

<table>
<tr><td><b>Course Code:</b></td><td colspan="3">CSC406S6</td></tr>
<tr><td><b>Course Title:</b></td><td colspan="3">Research Project</td></tr>
<tr><td><b>Credit Value:</b></td><td colspan="3"><b>06</b></td></tr>
<tr><td rowspan="2"><b>Hourly Breakdown:</b></td><td><b>Mentoring</b></td><td><b>Practical</b></td><td><b>Independent Learning</b></td></tr>
<tr><td>20</td><td>--</td><td>580</td></tr>
<tr><td colspan="4"><b>Objectives:</b></td></tr>
<tr><td colspan="4">Develop capability of carrying out scientific research in the computing domain for solving real world problems.</td></tr>
<tr><td colspan="4"><b>Intended Learning Outcomes:</b></td></tr>
<tr><td colspan="4">• Identify a hypothesis and/or a research problem
• Formulate the detailed problem statement
• Frame a solution with appropriate research methodology
• Validate the proposed solution
• Perform scientific communication
• Defend the performed research, results and findings</td></tr>
<tr><td colspan="4"><b>Course Outline:</b></td></tr>
<tr><td colspan="4">• Engage in an academic year long research project under the guidance of academic supervisor(s) with optional guidance by an external mentor
• Develop a research proposal with adequate literature review
• Carry out the research using appropriate research methodology
• Document and present the research</td></tr>
<tr><td colspan="4"><b>Teaching and Learning Methods:</b></td></tr>
<tr><td colspan="4">Reading assignments in journals, Research seminars, Open Educational Resources, Documentation</td></tr>
<tr><td colspan="4"><b>Assessment Strategy</b></td></tr>
<tr><td colspan="4">• Presentation of research proposal 20%
• Proposed solution and Implementation 40%
• Project Diary 10%
• Project report 10%
• Viva voce 10%
• Submission of abstract/poster/paper to a scientific forum 10%</td></tr>
</table>

Course Code:	CSC407S6		
Course Title:	Industrial Training		
Credit Value:	06		
Hourly Breakdown:	**Mentoring**	**Practical**	**Independent Learning**
	20	--	580

Objectives:

Provide an opportunity to develop skills and attitude, and gain experience in finding IT solutions to problems in an industrial environment.

Intended Learning Outcomes:

- Apply acquired knowledge in industrial environment
- Develop interpersonal, communication, management and team working skills
- Adapt to work readily in real industrial projects
- Perceive state-of-the-art industrial technologies

Course Outline:

- The industrial training is offered during the second semester.
- Students will be trained in an appropriate industry for a period of four to six months which amounts to 600 notional hours under the guidance of academic and industrial mentors.
- Any additional stay at the industry will not carry any additional credits.
- It is the student's responsibility to find a placement in consultation with the department. The department may assist the student in finding a placement in an appropriate industry.
- Students shall maintain a journal to record their progress activity during the training.
- Academic staff will visit the training institution at least once during the training period to monitor their progress.
- On completion of the industrial training, each student shall submit the report, journal and deliver an oral presentation.
- Students who fail to obtain a minimum grade of D+ in industrial training may opt for a general degree in Computer Science as the training is non-repeatable.

Training Methods:

Mentoring, Weekly recording of training diaries, Code reviews, Progress meetings, Supervised study

Assessment Strategy

• Training journal	20%
• Progress as per feedback from mentor(s)	20%
• Final Report	20%
• Presentation	40%

Course Code:	CSC403S3		
Course Title:	Data Science		
Credit Value:	03		
Hourly Breakdown:	**Theory**	**Practical**	**Independent Learning**
	30	**30**	**90**

Objectives:

Provide theoretical and practical knowledge on data science for solving data-driven problems and improving research skills in data science.

Intended Learning Outcomes:

- Discuss fundamental concepts of linear algebra in relation to data science
- Analyse data-driven problems using probability and statistics
- Apply mathematical optimization techniques for solving data-driven problems
- Build neural networks using backpropagation algorithm
- Transform data-driven problems into computer programs
- Analyse data through visualization

Course Contents:

- Linear algebra: Overview of scalars, vectors, matrices, tensors, multiplication of matrices and vectors, norms, trace, rank, eigenvalues, and eigenvectors
- Probability and statistics: Overview of probability, marginal and conditional probabilities, independence and conditional independence, probability density functions, expectation, variance and covariance, and Bayes rule; Bayesian classifier, principal component analysis (PCA), linear discriminant analysis (LDA).
- Information theory: Entropy, cross-entropy, KL divergence, mutual information
- Multivariate calculus and mathematical optimization for parameter estimation: partial derivatives, chain rule, gradient based optimization (Gradient Descent, Stochastic Gradient Descent), Jacobian and Hessian matrices, constrained optimization, neural networks and back propagation algorithm.
- Applications: Eigen faces, LDA based classification, Linear Regression, Least square minimization
- Emerging Technologies in Big Data Analytics: Usage of Open source frameworks (e.g. Hadoop).

Teaching and Learning Methods:

Lectures, Tutorials, Laboratory experiments, e-based teaching-learning, take home exercises, Simulations, Use of Open Educational Resources, Guided Learning

Assessment Strategy

In-course Assessment (Practical)	30%
End-of-course Examination	70%

References:

- Goodfellow, I., Bengio, Y. and Courville, A., Deep Learning, 1st Ed, MIT Press, 2016.
- Bishop, C.M., Pattern Recognition and Machine Learning, 1st Ed, Springer, 2006.
- Nielsen, M., Neural Networks and Deep Learning, Determination Press, 2019.
- Aggarwal, C. C., Linear Algebra and Optimization for Machine Learning, 1st Ed, Springer Nature, 2020