

UNIVERSITY OF CALCUTTA
Notification No.CSR/171/2002

It is notified for information of all concerned that the Vice Chancellor on 28.08.2002 has been pleased to approve the revised Syllabus of Computer Application for the 3 year B.Sc. (Major) course of study under this University as laid down in the accompanying pamphlet.

The above will come into effect from which academic session 2002-2003.

Ujjwal Kumar Basu

Registrar

Senate House
Kolkata: 700073

The 2nd September, 2002.

NOTIFICATION NO: CSR/171/2002

UNIVERSITY OF CALCUTTA
SYLLABUS FOR THREE YEAR B.SC (MAJOR) COURSE IN
COMPUTER APPLICATION

Scheme

N.B.: The number denotes marks.

Paper 1 – Theory

- A. Foundations of Computer Science–I: Discrete Mathematics (30).
- B. Programming Fundamentals–I: Introduction to Data Structure (30).
- C. Computer System Design–I: Basic Computer Architecture and Organisation (30).
- D. Computers and Social Sciences–I: Software Project Management (10).

Paper 2 – Practical

- A. Structured Programming Language–I: The C Programming Language (40).
- B. Scripting Languages–I (General): Python, Perl or Tel (20).
- C. Document Mark-up Languages: HTML and XHTML (20).

Sessional Work (20)

Paper 3 – Theory

- A. Programming Fundamentals–II: Introduction to Algorithms (30).
- B. Computer System Design-II: Operating System Concept (30).
- C. Applied Computer Science–I: Database System Concepts (30).
- D. Computers and Social Sciences–II: Open Source Movement (10).

Paper 4 – Practical

Programming Labs

- A. Structured Programming Language–II: The C++ Programming Language (40).
- B. Database Manipulation Language: SQL (20).
- C. Scripting Languages–II (Web): Web Based Client & Server Side Scripting (20).

Sessional Work (20)

Paper 5 – Theory

- A. Foundations of Computer Science–II: Introduction to Automata Theory, Languages and Computability (30).
- B. Applied Computer Science–II: Networks and Distributed Computing Concepts (30).
- C. Applied Computer Science–III: Software Engineering using OOA & D (30).
- D. Computers and Social Sciences–III: Computer Ethics and Netiquette (10).

Paper 6 – Practical

Programming Labs

- A. Structured Programming Language–III: The Java Programming Language (40).
- B. Symbolic, Functional & Logic Programming Language: Lisp/ML/Prolog (20).
- C. GUI Building using RAD Tools (20).

Sessional Works (20)

Paper 7 – Industrial Training & Internal Software Project

- A. Industrial Training (50).
- B. Internal Software Project (50).

Paper 8 – Entrepreneurship Development (100)

Paper 1(Theory)

A. Foundations of Computer Science–I: Discrete Mathematics (30)

Sets and Propositions: Combinations of Sets, Finite and Infinite Sets, Uncountably Infinite Sets, Mathematical Induction, Principle of Inclusion and Exclusion, Multi-Sets, Propositions. (3 periods)

Permutations, Combinations and Discrete Probability: Rules of Sum and Product, Permutations, Combinations, Generation of Permutations and Combinations, Discrete Probability. (4 periods)

Relations and Functions: Binary Relations, Equivalence Relations and Partitions, Partial Ordering relations and Lattices, Chains and Anti-chains, Functions and the pigeon-hole principle. (4 periods)

Graphs and Planner Graphs: Multi-graphs and Weighted Graphs, Paths and Circuits, Shortest paths in weighted graphs, Eulerian paths and circuits, Hamiltonian paths and circuits, The travelling salesman problem, Factors of a graph, Planner graphs. (4 periods)

Trees and Cut-sets: Trees, Routed Trees, Path lengths in routed trees, Prefit codes, Binary search trees, Spinning Trees and cut-sets, Minimum spinning trees. (4 periods)

Analysis of Algorithms: Time complexity of algorithms, Complexity of problems, Introductory remarks on tractable and intractable problems. (6 periods)

Discrete Numeric Function and Generating Functions: Manipulations of Numeric Functions, Asymptotic behaviour of numeric functions, Generating functions. (4 periods)

Recurrence Relation and Recursive Algorithms: Recurrence relations, Linear Recurrence relations with constant coefficients, Homogeneous solutions, Total solutions, Solutions by the method of generating functions. (4 periods)

Reference: Elements of Discrete Mathematics by C.L.Liu (TMH)
Chapter 1, 3, 4, 5, 6, 8, 9 and 10.
Discrete Mathematics by Johnson-Baugh (PE)
Discrete Mathematical Structures by Kolmann et al. (PHI)

B. Programming Fundamentals–I: Introduction to Data Structure (30)

Arrays: Abstract data types, Arrays as an ADT, Implementations of arrays, One dimensional and Multi-dimensional array, Sparse arrays. (3 periods)

Stacks: Stacks as an ADT, Implementations of stacks, example — Infix, postfix and prefix, Recursion, Simulating recursion using stacks. (4 periods)

Queues and Priority queues: Queues as an ADT, Implementations of queues and priority queues. (2 periods)

Linked Lists: Single linked lists, Double linked lists, Circular lists, Implementations of these. (4 periods)

Trees: Binary trees, Implementations of binary trees, example — Huffman algorithms, Tree traversals. (3 periods)

Sorting: Insertion, Selection, Bubble, Quick and Merge sort. (4 periods)

Searching: Binary search trees, B trees — only description. (4 periods)

Hashing: Implementation of hashes, Hash functions – perfect hash functions and universal classes of hash functions. (2 periods)

Graphs: Definitions of directed, undirected, acyclic graphs, paths, cycle, connectivity and bi-connectivity, Graph traversals, example — Minimum spanning tree, Kruskal's and Prim's algorithm. (4 periods)

Reference: Fundamentals of Data Structures in C++ by Horowitz, Sahani and Mehta
Data Structure Using C and C++ by Tanenbaum et al. (PHI)
Data Structure and Algorithms by Aho, Hopcroft and Ullman (AW)
Data Structure and Problem Solving Using C++ by Weiss (AW)

C. Computer System Design-I: Basic computer Architecture and Organisation (30)

Digital Logic Circuit: Logic gates, Boolean algebra, Map simplification, combinatorial circuits, Flip-flop-SR, D, JK, T flip-flops, Sequential circuits. (6 periods)

Digital Components: Decoders, Multiplexers, Registers, Shift registers, Binary counter, Memory units. (6 periods)

Digital Representation: Octal, Hexadecimal, Decimal and alpha-numeric representation, 1's complement and 2's complement, Fixed point representation, Floating point representation. (6 periods)

Micro operations: Binary adder, Binary adder-subtractor, Logic micro operations, Shift micro operations, Arithmetic Logic Shift unit. (6 periods)

Basic Computer Organisation and Design: Instruction codes, Computer registers, Computer instructions, Timing and control, Instruction cycle and pipeline, Design of basic computer – control logic gates, control of register and memory, control of single flip-flops, control of common bus. (6 periods)

Reference: Computer System Architecture by M.Morish.Mano (PHI)
 Chapter 1 to 5.
Structured Computer Organisation by Tanenbaum (PHI)

D. Computers and Social Sciences-I: Software Project Management (10)

Programming Style and coding Standards. (1 period)

Introduction to Software Project Management. (2 periods)

The dependence of time of completion on the quality, count and constitution Of project team – inadequacy of man-month concept. (2 periods)

Organisation of Software Projection: Statement of mission, Role of producer/manager, Role of technical directory/architect, Schedule, Division of labour, Interface definitions of modules. (2 periods)

Core-principles in good software management. (2 periods)

Introduction of industrial Standards such as ISO 9002. (1 period)

Reference: The Mythical Man-Month by Brooks (PE)
Programming Pearls by Bentley (PE)

The Practice of Programming by Kernighan (PE)
GNU Coding Standards (E-text)
File: Hierarchy Standard (E-text)

Paper 2(Practical)

Programming Labs

A. Structured Programming Language–I: The C Programming Language (40)

Constants, Types, Expressions, Statements, Functions and Procedures, Arrays (single and multi-dimensional), Structures and Unions, Pointers, Input/output.

Prime numbers generations.

Sum of Series calculation, Calculation of $\exp(x)$, $\log(x)$, $\sin(x)$, $\cos(x)$ to arbitrary many decimal digits, Calculations of factorials of large numbers.

Newton Raphson's method of finding roots, Gauss's elimination method for solving simultaneous equations, Gauss-Siedel method for finding roots, Simpson's method for numerical integration, linear regression, Polynomial fitting.

Knapsack and Towers of Hanoi problem.

Minimum spanning tree of a graph, Transportation and assignment problem.

At least 10 lab assignments.

Independent Programming Project in C–Complex Data structure implementation.

Reference: The C Programming Language by Kernighan and Ritchie (PHI)
Numerical Recipes in C by W.H.Press et al.
Operations Research by H.Taha.

B. Scripting Languages–I (General): Python/Perl/Tel (20)

Types, Expressions, Statements and Functions.

Scripting language lab problem should concentrate mainly on input/output filters, analysis of input files and summary generation, structured input parsing, and configuration files manipulation.

At least 5 lab assignments.

Reference: Python — Programming Python by Lutz (O'Reilly)
 Perl — Programming Perl by Larry Wall (O'Reilly)
 Tel/TK — Tel/TK Toolkit by John E. Ousterhout (PE)

C. Document Mark up Languages: HTML and XHTML (20)

Simple web design in all basic elements including tables except forms.

At least 5 lab assignments.

Reference: HTML & XHTML The definitive Guide by Musciano et al. (O'Reilly)

Sessional assignments (20)

Home assignments (5)

Programming assignments on Data Structure. (5)

Torn paper of software Project management. (5)

Term paper on current developments: Fuzzy logic, Neural Networks, Virtual Reality, Data Mining, Embedded systems, Computer Graphics, Animation, Multimedia and other similar current topics. (5)

Students should become familiar with a suitable office suite similar to Microsoft Office, Star Office, Lotus Smart Suite or Open Office.

Paper 3(Theory)

A. Programming Fundamentals-II: Introduction to Algorithms (30)

Mathematical foundation: Introduction, Growth of functions, Summations, Recurrences. (4 period)

Sorting and Order Statistics: Heap sort, Quick sort, Sorting of linear time, Medians and order statistics. (5 periods)

Design and analysis technique: Divide and conquer, Dynamic programming, Greedy algorithms, Backtracking – Alpha-beta pruning & branch and bound, Local search. (4 periods)

Advanced Data Structure: Red-black tree, B-Trees-Implementation. (4 periods)

Graph Algorithms: Elementary graph algorithm, (Breadth first search, Depth first search, Topological sort and connected components), Minimum spanning trees, Shortest paths. (10 periods)

Computational geometry: Convex hulls, Closest pair or points. (2 periods)

Reference: Introduction to Algorithms by Corman et al (PHI)
Chapter 1,2,3,4,7,8,9,10,14,16,17,19,23,24,25,26 and 35
The Design and Analysis of Computer Algorithms by Aho et al (PE)
The Art of Programming by Knuth (3 volume) (PE)
A Bundle of Algorithms in C++ by Sedgwick (5 volume) (PE)

B. Computer System Design-II: Operating System Concept (30)

Structure of operating systems: System components, Operating system services, System calls, System programs, System structures, Virtual machines. (2 periods)

Processes and threads: Concept, Operations on processes, Cooperating processes, Threads, Inter-process communication. (3 periods)

Scheduling: Concepts, Scheduling criteria, Scheduling algorithms, Multi-processor scheduling, Real-time scheduling. (3 periods)

Process synchronizations: Concepts, Critical-section problem and its software and hardware solution, Semaphores, Monitors, Automatic transactions. (3 periods)

Deadlocks: Characterization, Prevention, Avoidance, Detection, Recovery. (3 periods)

Memory management: Logical vs. physical address space, Swapping, Paging, Segmentation. (3 periods)

Virtual memory: Demand paging, Page replacement, Thrashing, Demand segmentation. (3 periods)

File system: Concept, Access method, Directory structure, Allocation, Free-space management. (3 periods)

Input/output devices: Disk structure, Disk scheduling, Disk management. (2 periods)

Protections: Goals of protection, Domains, Access matrix, Capability based systems. (2 periods)

Security: Concept of security, Authentication, Program threats, System threats, Encryption. (2 periods)

Case studies – UNIX, Mac and Windows. (2 periods)

Reference: Operating System Concepts by Silberschatz et al. (Wiley)
Modern Operating Concepts by Tanenbaum (PHI)
Operating System: Internals and Design principles by Stallins (PHI)

C. Applied Computer Science I: Database System Concepts (50)

Introduction database theory: Data model – Entity – Relationship model, Relational model, Network model, Hierarchical model, Object-oriented model, Data definition language, Data manipulation languages, Database manager, Database administrator, Database users. (5 periods)

Relational database SQL: Query by example, Integrity constraints, Domain constraints, Referential constraints, Functional dependencies, Assertions, Triggers, Relational-database design – Normalization, 1NF, 2NF, 3NF, 4NF. (6 periods)

Object based Database and XML: Object-oriented databases, Object-relational databases, XML. (6 periods)

Data storage and Querying: Storage and file structure – file organisation, sequential files, data dictionaries, buffer management, Indexing – B-Tree, B+-tree, Hashing – static and dynamic, Query processing, Query optimization.(6 periods)

Transaction management: Transaction – storage model, deadlock handling, Transaction definition in SQL, Concurrency control – Schedules, Serializability, Lock based protocols, Time-tamp based protocols, Validation techniques, Multiple granularity, Multi-version schemes and recovery system. (3 periods)

Security and Integrity: Security and Integrity violations, Authorization and views, Security specification in SQL, Encryption. (3 periods)

Reference: Database System Concept by Silberschatz et al. (MH)
 Chapter 1 to 17
Database System: The complete book by Ullman.

D. Computers and Social Science II : Open Source Movement (10)

Hacking vs. Cracking. (1 period)

Brief History of Hacking. (1 period)

Definition of Open Source Software. (1 period)

The concept of Open Source, Pros and cons of open source software, Pros and cons of proprietary software. (2 periods)

The Cathedral and Bazaar models of software development. (2 periods)

Case Studies – BSD, GNU, Linux. (1 period)

Salient features of different types of licences – open and proprietary. (1 period)

Future of open source movement. (1 period)

Reference: Open Source: Voice from the Open Source Revolution edited by Mark Stone et al. (O'Reilly)
The Cathedral and The Bazaar by Eric S. Raymond. (O'Reilly)
Halloween Documents I to VI annotated, edited or written by Eric S. Raymond. (E-text only)

Paper 4(Practical)

Programming Labs

A. Structured Programming Language–II: The C++ Programming Language (40).

Classes and objects.
 Modularisation – Namespaces
 Data abstraction – interface and implementation.
 Data encapsulation- public, protected, private declarations.
 Inheritance – Base classes and derived classes.
 Polymorphism – virtual functions.
 Constructors, Destructors, copy constructors.
 Operator overloading.
 Templates – class and function templates.
 At least 10 lab assignments.

Independent programming project in C++ – Complex Algorithm Implementation (e.g. Capacitated Flow Network/Polygon Triangulation/Graph Planarity Testing/Travelling Salesman Problem/Graph Isomorphism/Maze Searching/Time Table Scheduling/Project Scheduling Assignment/Transportation).

Reference: The C++ programming Language by B.Stroustrup (AW)
The Annotated C++ Reference Manual by Ellis

B. Database Manipulation Language: SQL (20).

SQL SERVERS – Simple Forms, Queries and Reports using a SQL Server (Postgres/Oracle/Microsoft SQL Server/MySQL Server)

At least 5 lab assignments.

Independent database programming project – A database application (e.g. Library/Financial Accounts/Student (Fee or marks)/Company Stock and Inventory/Sales/Personal Databases).

Reference: SQL in a Nutshell by Kline et al. (O'Reilly)
Postgres SQL by Bruce Momjian (AW)

C. Scripting Languages–II (Web): Web Based Client & Server Side Scripting (20).

Client side scripting (JavaScript)

Server side scripting including CGI (PHP/Python/Perl)

At least 5 lab assignments.

Independent web programming project – A server-client application (e.g. Guest book log/Web ordering form/Remote database management/Remote software configuration/Chat/Web news/Web mail).

Reference: JavaScript: The Definitive Guide by Flanagan (O'Reilly)

Sessional Assignments (20)

Home Assignments. (5)

Programming assignments on algorithms, Operating system (UNIX signals, Threads, Say VIPC, Files operations, Forks and pipes in UNIX or Windows API in Windows). (5)

Term paper on open software. (5)

Independent programming project on 2D graphics programming – OpenGL/DirectX/Windows/X-Windows, Multimedia Programming (e.g. a simple RPG/RTS/FPS game or an Educational Multimedia Presentation)). (5)

Reference: Advanced programming in the Unix Environment by R Stevens (PE)
Linux Programmer's Guide (LPG) from Linux Documentation Project
Interactive computer graphics: A top down approach with OpenGL by Angel et al.
Computer Graphics: Principal and Practices in C by Floy et al.
Computer Graphics, C version by Hearn and Baker

Paper 5(Theory)

A. Foundations of Computer Science–II: Introduction to Automata Theory, Languages and Computability (30).

Central concepts of Automata Theory: Alphabets, Strings, Languages and problems. (2 periods)

Finite Automata: Deterministic finite automata, Nondeterministic finite automata, Equivalence of Deterministic finite automata and nondeterministic finite automata, Finite automata with epsilon-transitions, Elimination of epsilon-transitions. (4 periods)

Formal languages: Introduction, type-0, type-1, type-2 and type-3 languages. (2 periods)

Regular sets and regular grammars: Regular expressions, Operators of regular expression, Finite automata and regular expressions – Converting deterministic finite automata to regular expression, Converting regular expressions to finite automata, Algebraic laws for regular expressions, Pumping lemma for regular languages, Statement and applications, no proof. (4 periods)

Context free languages: Context free grammars – Definition, Derivations using a grammar, Parse trees, Ambiguity in Grammars and Languages, Pumping lemma for context free languages, Statement and applications, no proof. (4 periods)

Pushdown automata: Definition, Equivalence of pushdown automata and context free grammars, Deterministic pushdown automata. (4 periods)

Turing machines and linear bounded automata: Definition, The language of Turing machine, Halting problem, Equivalence of one-tape and multi-tape Turing machine, Non-deterministic Turing machine. (4 periods)

Reference: Theory of Computer Science (Automata, Language and Computation) by Mishra and Chandrasekharan. (PHI)
Introduction to Automata Theory, Language and Computation by Hopcroft, Motwani and Ullman.
Elements of the Theory of Computation by Lewis and Papadimitriou.

B. Applied Computer Science-II: Network and Distributed Computing Concepts. (30)

Evolution: Single user systems to networks, peer-to-peer and client-server concepts, File server and application server. (2 periods)

LAN/MAN: bus, tree, star, ring topology, Transmission media-UTP, coaxial and optical fibre, wireless transmission, Circuit switching and packet switching, Network devices, NICS, modems, hub, repeater, bridge, router and gateway, WAN network of networks. (2 periods)

Internet: basic idea, IP addresses and classification, Basic idea of TCP/IP, E-mail, Usenet, Telnet, FTP, WWW and browsers, accessing and downloading. (2 periods)

Data encoding: Amplitude, frequency and phase shift keying. (2 periods)

Unix Communication protocols: TCP, UDP. (2 periods)

Berkley sockets: TCP and UDP sockets-socket-connect, bind, listen and accept functions, TCP Client-server example-TCP Echo server and client. Input/output multiplexing using select and poll system functions, UDP client-server example- UDP echo server and client. (4 periods)

Name and address conversion: get host by name, get host by address functions. (2 periods)

Daemon processes and instead super server: Syslogd daemon, syslog function, daemon-init function, instead daemon, daemon-instead function. (4 periods)

Non-blocking I/O using non-blocking connect and accept. (2 periods)

Unicasting, Broadcasting and multicasting in LAN/MAN. (2 periods)

Threads: TCP Echo server using threads. (2 periods)

Inter-process communication using Remote procedure calls. (4 periods)

Reference: Computer Networks by A.S.Tanenbaum
Data and Computer Communication by William Stallings
UNIX Networks Programming (Vol. I and II) by Richard Stevens (PE)
TCP/IP Illustrated (3 volumes) by Richard Stevens (PE)
Computer Networking: A Top Approach Featuring the Internet by Kurose (AW)
Internetworking with TCP/IP (3 volumes) by Comer et al.

C. Applied Computer Science-III: Software Engineering Using object oriented analysis and design (30)

System development models: Waterfall or linear cycle, Staged development, Prototyping, Evolutionary development or spiral model. (1 period)

Stages of system development: Feasibility study, Requirement analysis, System analysis, Structured system design, System development, System validation, System testing, System maintenance, Documentation. (3 periods)

Introduction of object-oriented analysis and design: The object model, Classes and objects, Classification of classes. (2 periods)

The method of object-oriented analysis and design: Class diagrams, State transition diagrams, Object diagrams, Interaction diagrams, Module diagrams, Process diagrams. (12 periods)

The Micro and Macro development process. (6 periods)

Unified modelling language introduction. (4 periods)
Case studies. (2 periods)

Reference: Object-oriented Analysis and Design with Supplifications by Grady Booch. (AW)
Object-oriented Modelling and Design by Rumbaugh. (AW)
The Unified Modelling Language User Guide by Booch. (AW)
The Unified Modelling Language Reference Manual by Rumbaugh. (AW)
Introduction to System Analysis and Design by Hawryszkiewycs.
Software Engineering: A practitioner's Approach by Pressman.
System Analysis and Design by Kendall and Kendall.

D. Computer and social science-III: Computer ethics and netiquettes. (10)

Introduction to computer ethics. (1 period)

Philosophical and professional ethics. (2 periods)

Ethics and the internet: Social implications and social values. (1 period)

Netiquette basics: Core rules of netiquette, Netiquette for electronic-mails, Netiquette for discussion groups, Rules of flaming, Netiquette for information retrieval, Violations of netiquette. (2 periods)

Business netiquette: Electronic mail at work, Netiquette related to company and cyberspace. (1 period)

Social netiquette: Netiquette at home and school. (1 period)

Reference: Computer Ethics by Johnson (PE)
Netiquette by Virginia Shea (E-text also available)
Computer Ethics and Social Values by Johnson and Nissenbaum (AW)
A gift of Fire: Social, Legal and Ethical issues for computers and the Internet by Sara Baase. (AW)

Paper 6(Practical)

Programming labs:

A. Structured Programming Language-III: The Java Programming Language (40)

At least 10 lab assignments.

Independent programming project on Java-A client-server application (Network conference/Railway reservation/Bank accounts/simple network game).

B. Symbolic, Functional & Logic Programming Language: Lisp/ML/Prologue (20)

At least 2 lab assignments in each language-Very basic programs.
Independent programming project on Lisp, Prologue or ML-A simple expert system.

C. GUI Building using RAD Tools (20)

At least 5 lab assignments.

Independent programming project— A configuration tool for any application.

Reference: Java in a Nutshell by Flanagan (O'Reilly)
Core Java (2 volumes) by Horstman et al. (FX)
Programming languages by Ravi Sethi (AW)
Concept of programming languages by Sebesta (AW)
Lisp by P.H.Winston and B.K.Paul Horn (AW)
Golden Lisp by C.L.Steel.

Seasonal assignment (20)
Home assignment (5)

Programming Assignment of UNIX Tools e.g. Flex, Bison, Network and Distributed (Server/Client) Programming using BSI Socket/HC/CORBA/COM/NET/JINI (5)

Term paper on Computer Ethics. (5)

Assembly, Installation(Linux/Windows-Partitioning and setup), Configuration (Local Area network, web Server, Network file Server, Network Print Server, Web Server, Mail server, Password Server, Firewall, Proxy Server, news Server, PTP/Telnet Server, Domain Name Server, DHCP Server) and Maintenance (Backup, File system Consistency Checking, Defragmentation) of PCs. (5)

Reference: System Administrator's Guide (SAG) and Network Administrator's Guide (NAG) for Linux documentation project.

Paper 7

Industrial Training and Internal Software Project

A. Industrial training (50)

Industrial training should be of at least 2 months duration during the 3rd year at any IT concern/organisation which will give a report at the end of training.

The students are to submit at least 2 copies of the industrial training report to this college at the time of Part-II examination. The report should be duly certified by the concerned authorities of the organisation where they underwent training. There will also be an oral presentation of the report by the students followed by a viva-voce examination.

B. Internal software project(50)

The student has to give an oral presentation on the internal software project followed by viva-voce during the Part-II examination.

The students are supposed to submit 2 copies of the project reports to the college. The report should be duly certified by the project guide and the concerned authorities of the college.

It is suggested that the students do their internal software project on current topics of public utility.

General programming guidelines for Linux/Posix platform.

For UNIX projects the preferable platform will be Linux, GNU-Hurd, Free BSD, Solaris or Cygwin. The students should preferably do the project maintenance using Makefile, version control using PCS/ CVS/SCCV, automatic configuration script generation using Autocentf, software configuration should be by start-up scripts and documentation using SGHL Tools/Dockbook (preferred) or Latex. Software extensibility and modularization, if required, might be provided using embedded interpreters (guide or python). Programming language should be preferably C/C++/Java/Sisp/Prolog/SQL whichever is suitable. It is suggested further the portability should be of first concern unless programming specifically for a particular platform or suite of product. For C or C++ either GNU coding standards or Linux kernel coding style (essentially same as Kernighan and Ritchie) and for other languages acceptable coding standards can be followed by students. For other purpose such as versioning, program behaviour and documentation, the program may suitably adhere to GNU standards and Dehiad policies.

General programming guidelines for Windows platform.

Proper project maintenance and version control could be provided for by the software application by using appropriate tools. Documentation should be done preferably in HTML and programming package should preferably be Visual Studios/Borland's C++ Builder/Delphi RAD platforms.

Paper-8

Entrepreneurship Development (100)

The syllabus of Entrepreneurship development is left unchanged to remain compatible with other streams having it is uncommon paper.

Recommended setup of Computer Laboratory:

At least one computer per student at the time of practical and exams should be made available. Similarly, for the required software packages, there should be at least one package per student at the time of practical and exams. A LAN is highly recommended. A printer to the LAN for every student will suffice the needs.

It is suggested to leave at least a few Intel processor computers of current technology in the lab. All these are recommended to be dual booting in Linux and Windows. It is also recommended that Linux be installed as a development system with as many development libraries and tools as possible.

Internet connectivity is recommended for downloading useful public domain software and libraries from source code repositories and also needed reference documentation.

A CD-Writer and a scanner are also recommended.

Operating System:

Debian/Redhat/Suse/Mandrake GNU-Linux

Windows 95/98/NT/XP/NT

Debian Hurd

Solaris

FreeBSD/NetBSD/OpenBSD

GNU/Linux, Gnu ECGS (C&C++)/Mingw (also for cross platform development) Gnu-Gej (Java compiler)

Clisp/Cuile/CMU Lisp/Can Prolog/GCL/Ocaml

Postgres/MySQL/Oracle
 X Windows
 GTE/Motif/Athena/Openview Toolkit GUI Development Packages
 Apache/Zope web server with PHP/Python
 MesaGL 3D graphics development library
 Flex, Bison
 Make, CVS/SCS/SCCS, autoconf, automake and other GNU system utilities
 Open Office/Star office/Tetex
 Mozilla/Netscape navigator/Opera
 Delehi/Kylix
 Wine
 Gimp
 Windows
 Microsoft VC++/Mingw/Cygwin/Borland C++/Delehi/Kylix
 Perl/Python/Tel
 Golden/Sybase/Microsoft SQL Server.
 Visual Studio
 IIS Web Server
 DirectX 3D graphics development Library
 Star Office/Lotus SmartSuite/Microsoft Office
 Netscape Navigator/Internet Explorer/ Opera
 10 General Notes and instructions

The distributions of marks for the lab examination papers will be as follows: lab examination problems 10 (there will be 3 problems with marks divided as 20+10+10 proportioned to periods allotted), Viva-voce 20, Discussion 10, Lab/project reports 10, Sessional work/term papers 20. The duration of lab examinations shall not exceed 6 hours including suitable time for discussion and viva-voce.

It is suggested that at least the 1st year labs be preceded by a theoretical discussion of the C programming language (around 10 periods). The other language labs may require only a minimum discussion. It is further suggested that Data Structure be started after these preliminary classes of the C programming languages. The students are encouraged to pick up as many popular general purpose computer programming languages as possible, on their own during the 3 year course.

It is available that the students complete all their laboratory work as well as term papers and projects before the commencement of their theory exams on various subjects.