

PRINCIPAL SUBJECT AREA

COMPUTER SCIENCE

100 LEVEL COURSES

CS 100 Computer Applications (2 credits)

Introduction to Computer and operating Systems,

Micro Computer Applications: Use of Software Packages- Spread Sheet applications, DBMS applications, Utility programs and Word processing.

Data Protection Techniques: Data security techniques, Computer Viruses and prevention. Data Communication: Email, Internet and Networking of Computers.

Introduction to a Programming Language: Procedures, Functions, File handling, Application of a DB management.

(This course includes both theory and practicals)

Recommended Texts:

1. Computer Science, C.S.French
2. Programmer's Guide to Foxpro 2.0, D. Howard
3. Computer viruses, Robert Slade

CS 101 Introduction to Computer Science (3 credits)

Introduction and overview: Intelligent machines and systems applications, Business, Communications, Educational, Engineering, Environmental, Medical and Scientific applications.

Introduction to computing concepts: Basics of computer programming: data types, declarations, assignments, basic input and out put ASCII files, built-in functions.

Structured programming ideas: selection statements: sequence, iteration (counting loops, while loops, file pointers), conditional (if-then-else statements, case statements), matrix manipulations (addition, subtraction, multiplication, transposition).

Modular programming: functions, procedures with actual and formal parameters, simple sort algorithms, dynamic memory allocation and addressing.

Numerical methods: Linear interpolation, linear regression, pseudo random, roots of functions, solutions of simultaneous linear equations by Gaussian elimination, numerical integration.

Recommended Texts:

1. The Thinking Ape: Evolutionary Origins of Intelligence, R. Byrne.
2. Intelligent Multimedia System : A Handbook for Creating Applications, R.M.Kaplan
3. Artificial Intelligence, E.Rich and K. Knight

CS 102 Programming Techniques (3 credits)

Basic concepts, basic components of programming languages, binding, simple algorithms operating on non-structured data, modularity in program construction.

Basics of constructing larger programs :abstraction and instantiation of program components, structured data (lists, stacks, queues, ordered binary trees), storing and accessing data structures, operations on mutable data, working with mutable data, object-based programming, data encapsulation

Recommended Texts:

1. *Data Structures, Algorithms, and Object-oriented Programming*, G.L. Heileman.
2. *Structured programming concepts*, K. Labudde

CS 104 Structured oriented Programming practical (1 credit)

(Prerequisites: CS101, CS102)

Language constructs: data declarations, loops, decision structures, input/output, files, subprograms / procedures, numeric and non-numeric 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

Recommended Texts:

1. *The C Programming Language*, 2nd Edition, by Brian W. Kernighan and Dennis M. Ritchie, Prentice Hall, Inc., 1988.

CS 105 Object oriented Programming practical (1 credit)

(Prerequisites: CS101, CS102)

Implementation of programs with object oriented language constructs: classes, objects, inheritance, aggregation, composition and polymorphism.

Recommended Texts:

1. *Developing Java Software*, 3rd Edition, by Russel Winder and Graham Roberts, published by John Wiley and Sons, 2006
2. *Java Programming: From the Beginning*, K. N. King, Georgia State University

<http://www.pdn.ac.lk/sci/>