INSTITUTE OF HUMAN RESOURCES DEVELOPMENT
Prajo'e Towers, Vazhuthacaud, Thiruvananthapuram, Kerala, India. Pin695 014
http://www.ihrd.ac.in

POST GRADUATE DIPLOMA IN COMPUTER APPLICATIONS

(Two semesters)

Scheme & Syllabus

2018

(Effective from January 2018 admission)

INSTITUTE OF HUMAN RESOURCES DEVELOPMENT

POST GRADUATE DIPLOMA IN COMPUTER APPLICATIONS

(Two semesters)

Subjects of study and Scheme of Assessment
(Scheme-2018)

First Semester

Subject Code	Subject Name	No. of Hrs/Week		Minimum Marks			Maximum Marks		
		T	P	W/P	CE	T	W/P	CE	T
PGDCA101	Computers and Information Technology	3	-	40	20	75	100	50	150
PGDCA102	Computer Networks & Internet	3	-	40	20	75	100	50	150
PGDCA103	Programming Techniques	3	-	40	20	75	100	50	150
PGDCA104	Software Development & Databases	3	-	40	20	75	100	50	150
PGDCA105	Lab Practice - 1 *(C++, Python)*	-	3	40	20	75	100	50	150
PGDCA106	Lab Practice- 2 (PC Skills & Database)	-	3	40	20	75	100	50	150
Duration : Total 300 Hrs.		12	6	TOTAL MARKS			600	300	900

Second Semester

Subject Code	Subject Name	No. of Hrs/Week		Minimum Marks			Maximum Marks		
		L	P	W/P	CE	T	W/P	CE	T
PGDCA201	Advanced Web Technologies	3	-	40	20	75	100	50	150
PGDCA202	Programming in Java & Android	3	-	40	20	75	100	50	150
PGDCA203	Advances in Information Technology	3	-	40	20	75	100	50	150
PGDCA204	Lab Practice - 4 (ASP.Net, PHP)	-	3	40	20	75	100	50	150
PGDCA205	Lab Practice - 5 (Java & Android)	-	3	40	20	75	100	50	150
PGDCA206	Project work	-	3	40	20	75	100	50	150
Duration : Total 300 Hrs.		9	9	TOTAL MARKS			600	300	900

** T- Theory P - Practical W - Written CE–Continuous Evaluation T – Total*

[Scheme 2018]

PGDCA101 Computers and Information Technology

Module 1: Functional units, classification of computers, desktop, laptop, tablet etc.,concepts of processor-survey of current processors, memory-primary and secondary, survey of primary memory modules, secondary storage devices-different types, hard disk technologies etc.. Flash memory, optical storage devices, capacity, use etc. I/O devices-scanner, camera, printer, biometric devices. Display devices, Comparison of LED and TFT displays. Anatomy of desktop computer system. UPS- online & Offline, Specification of a typical PC (desktop, laptop). (12Hours)

Module 2: Software- Concepts, system software, application software, proprietary &open source, operating systems, windows, linux- basic features and operations of both windows and Linux- comparison. Programming languages – high level vs low level, assembler, compiler, interpreter. Different types of programming languages-survey,desktop publishing software, image editing tools, Databases (12Hours)

Module3: Information Technology: Data, information, Introduction to number systems-decimal, binary, octal, hexadecimal – Conversion between number systems - representation of characters, integers, floating point numbers. Display - Pixels, resolution-HD/4K etc, - Survey of different file formats- JPG,GIF, PNG, PDF, XPS, MP3/MP4, AVI, etc. (basic concepts only). (12Hours)

Module 4: Applications of IT- education, media, entertainment etc, survey of application tools –photo editing - gimp, internet facilities-browser, search engine, online payment systems, E-cash, cashless payment, wallets, social networking,blog, online tools – online storage - virtual keyboards, google forms, calendar, mobile apps. Security-computer virus, antivirus, malware, worms. (14Hours)

References:

1. E.Balagurusamy, *Fundamentals of Computers*, McGraw hill
2. Dennis P Curtain, *Information Technology: The Breaking wave*, McGrawhill
3. Peter Norton, *Introduction to Computers*, McGrawhill, Seventh edition
4. Computer fundamentals, Architecture & organization – B RAM – New age international.
5. Computer fundamentals – Rajaraman – PHI

* * * * * * *

[Scheme 2018]

PGDCA102 Computer Networks & Internet

Module 1: Computer networks- LAN, WAN MAN, devices, topology. Basic terminology-band width, baud rate, switching basics. Topology: bus, tree star, mesh, ring. Transmission media: Guided- twister, coaxial, fibre cables. Unguided media- radio and microwave transmission. Devices: Modem, Hub, switch, NIC, repeater, bridge, router, Gateway. (10 hrs)

Module 2: Internetworking: www, internet access methods - DSL, Cable, wireless, internet address and domains - URL, web browsers, Website, web server, web hosting, different steps in hosting a website,concepts of TCP/IP, IPv4 and IPv6 comparison, DNS, web portal, e-mail servers, search engines and searching tips, Types of web documents – static, dynamic, responsive, EPN, VPN. – Concept of Cloud computing. (12 hrs)

Module 3: wireless networking:Mobile and wireless devices, Wireless transmission, multiplexing, modulation, Wireless cards, access points - bluetooth - Spread Spectrum, Cellular Systems, Medium Access Control, Basic concepts of Telecommunications System: CDMA and GSM, DECT, TETRA, UMTS and IMT 2000, Satellite System, Broadcast Systems, Wireless LAN: IEEE 802.11, Hyper LAN. Introduction to GPRS, comparison of 3G, 4G networks

(12 hrs)

Module 4: Concepts of HTML - Organisation of HTML document- HTML tags- Meta, Title, body - Headings, Paragraphs, Links,Images, Elements, Attributes, Comments, Line Breaks- Text Formatting-HTML Style Attribute. Links, Hyperlinks,Anchors. Image Tag- Embedding video/audio in HTML.Tables - Attributes. Lists- Ordered & Unordered Lists.HTML Forms and Input, Form Tags, Colors, Color Values, Color Names- Frames- Frame Tags, Fonts,attributes– Character Entities- The Head Element, URL. Introduction to Cascading Style Sheet (CSS), CSS Syntax, Comments, Id and Class, Text formatting- CSS Font - Font Families. Using Pixels and Em - CSS Lists- List Item Markers, Cross browser Solution - CSS Tables, Grouping and Nesting, Dimension- Display and Visibility- Block and Inline Elements, Positioning- Static, Fixed, Relative, Absolute Positioning. Overlapping Elements, Pseudo-classes, Image Gallery. Introduction to javascript. (16 hrs)

References:

1. Brijendra Singh, *Data Communication and Computer Networks*, 2/e, PHI
2. Behrouz A Forouzan, *Data Communication and Computer networks*, 4thed,McGraw Hill
3. Achyut S Godbole, *Data communications and networks*, McGrawHill, Second1.
4. Mastering JavaScript - BPB
5. JavaScript Bible IDG Books, Danny Goodman
6. CSS - The Definitive Guide, O'reilly Publications

* * * * * * *

[Scheme 2018]

PGDCA103 Programming Techniques

Module 1: Programming: Steps in program development – Algorithm, flowchart, Coding, Structure of a simple program-writing main(), a simple program, compiler- Types of errors, Linking and Executing - Concept of procedure oriented and object oriented languages.Introduction to C++, Identifiers and keywords – Variables - Data types – primary data types - Type Modifiers – long, short,signed and unsigned, Constants, - Variable declaration. Operators – Arithmetic, Relational, Logical, Conditional, comma operator, sizeof operator, Precedence of operators, Type casting. Library & header files. (8hrs)

Module 2: Control Structures – if else construct, nested if - switch statement. Concept of loop - While loop, for loop, do while loop, nesting of loops. Break, continue.Arrays - Concept of an array - Advantage - concept of dimension and size - Array declaration, Array Manipulations (accessing element, Entering data, Reading data, initialization) – printing. Two dimensional arrays – Character strings - Initializing character array - Null character, reading character strings. Concept of pointers - & and * operator - pointer expression – pointer assignment - pointer arithmetic - pointer comparison - Dynamic memory allocation - malloc(), calloc(), free() -command line arguments.Functions – Concept of a function - General form of a function - Function with arguments - return statement- default return type and the type void - formal arguments, actual arguments. Local and global variables (14 hrs)

Module 3: Object oriented Programming: Structures – Structure as a collection of data – syntax of declaring and Initializing structures, Accessing structure elements, Object oriented concepts – defining class, creating objects, Simple programs. Features of OOPS (Data abstraction, Data hiding, Data encapsulation,inheritance, polymorphism etc.)–constructor, destructor, friend function, function overloading, operator overloading.Inheritance-simple examples.File handling – Concept of a file - simple file based programs. (12 hrs)

Module 4: Introduction to Python: Python Versions, Installing Python, Environment Variables, Executing Python from the Command Line, IDLE, Editing Python Files, Getting Help, Dynamic Types, Python Reserved Words, Naming Conventions, Basic Python Syntax- Comments, String Values, String Operations, String Operators, Numeric Data Types, Conversions, Simple Input and Output, The print Function.
Control Flow and Syntax, Indenting, The if Statement, Relational Operators, Logical Operators, True or False, Bit Wise Operators, The while Loop, break and continue, The for Loop, Collections- Lists, Tuples, Sets, Dictionaries, Functions- Defining Your Own Functions, Parameters, Function Documentation, Keyword and Optional Parameters, User defined Functions. Classes in Python: Creating Classes, Instance Methods, File Organization, Special Methods, Class Variables, Input and Output- Data Streams, Creating Data Streams, Access Modes, Writing & Reading From/To a File. (16 hrs)

References:

1. Ashok N. Kamthane, *Object oriented Programming with ANSI & Turbo C++,* Pearson
2. H M Deitel and P J Deitel, C++*: how to program*, Pearson Education
3. Robert Lafore, *Object Oriented Programming in Turbo C*++, Galgotia Publications
4. Object Oriented Programming With C++ - Balagurusamy E. - Tata Mcgraw Hill
5. Beazley, D. M. (2009). Python essential reference. Addison-Wesley Professional.
6. Barry, P. (2010). Head First Python. " O'Reilly Media, Inc.".
7. Punch, W. F., &Enbody, R. (2010). The practice of computing using python. Addison-Wesley.
8. Mark, S. (2009). Programming in Python 3.Pearson Education India.
9. Lutz, M. (2013). Learning python." O'Reilly Media, Inc.".

* * * * * * *

[Scheme 2018]

PGDCA104 Software Development & Databases

Module 1: Introduction to Data base: database and DBMS -Managing Data, Concept of Database, Need for database, Advantages, characteristics, File vs database, data abstraction, independence, Users of Database-Database Administrator, Application programmers, Naive users-, Different levels of databases - internal, conceptual and external levels – schema. data models, Relational Databases - concepts of domain, tuple, degree, cardinality – keys -primary, secondary, candidate, foreign. Survey of popular DBMS, Concept of Big Data. (10 hours)

Module 2: Structured Query Language: Features of SQL, Creating a database, creating table, datatypes in SQL, SQL Commands: Creating tables, naming tables and columns, Constraints, Viewing the Structure of a table, Inserting data, Retrieving information, Aggregate functions, Eliminating duplicate values in columns, Selecting specific rows using WHERE clause, ORDER BY Clause, using GROUP BY Clause, Applying conditions, Modifying data in tables, Changing the structure of a table, Adding/changing/removing a new column, Renaming a table, Deleting rows from a table, Removing table from a database, Nested queries. Querying multiple tables - simple queries - nested queries - DCL statements - syntax and examples. (12 hours)

Module 3: Software Development: Characteristics of Software, Product and Process, Software Development Process models, Software Development Life Cycle Model: Waterfall Model, Prototyping, Comparison of different Life Cycle Models, Software Requirements Analysis and Definition: Requirements Specification: Need for SRS, Characteristics and Components of an SRS, Structure of a Requirements document. Functional Specification with Use cases, developing use cases, Structured Analysis, Planning a project, COCOMO model, quality plan.Function oriented design: Top-down and Bottomup Strategies, design issues, Interface design, forms, system design, Data Flow Diagrams,Developing the DFD Model of a system, Object-oriented design: Object-oriented design concepts, Unified Modelling Language, using UML, Class Diagram, Sequence Diagram, Collaboration Diagram; design walkthroughs, critical design review. (16 hours)

Module 4: Coding and testing: common coding errors, structured programming, coding standards, incremental coding process, test driven development, source code control and build, refactoring, code inspections, static analysis, unit testing, combining different techniques. Testing- error, fault and failure, test oracles, test cases, Black Box Testing, Equivalence Class Partitioning, Boundary Value Analysis, White Box Testing- control flow based and data-flow based testing, test plan, test case. Survey of CASE Tools. (12 hours)

References:

1. Rajib Mall, *Fundamentals of Software Engineering,* Second Edition, PHI
2. PankajJalote, *An Integrated Approach to Software Engineering*, Narosa
3. AtulKahate, *Introduction to Data Base Management Systems*, Pearson Education
4. Silberschatz· Henry F. Korth· S. Sudarshan, Database System Concepts McGraw-Hill
5. Bahrami, A., Object Oriented System Development using the Unified Modeling Language, McGraw-Hill
6. Jacobson, Ivar.,Object-oriented software engineering:b a use case driven approach. - Addison-wesley
7. Booch, Grady;;Rumbaugh , James; Jacobson, Ivar., The Unified Modeling Language user guide – Pearson Education Asia.
8. SQL in Ten Minutes, SAMS

* * * * * * *

[Scheme 2018]

PGDCA105 Lab Practice - 1

(C++,Python)

(Practice of concepts studied in PGDCA103 Programming Techniques. 50 Hrs.)

Programming practiceshallinclude the following topics:

1. Simple programs to practice and understand writing main(), using header files, dealing input and output, declaring variables, learning data types.
2. Programs to practice use of different forms of if-else construct (without arrays).
3. Programs to practice different forms of loops (while, for, do-while). (The same problem may be attempted using different loops).
4. Programs to learn arrays: declaration, reading data into arrays, performing computation on data within the array, displaying array contents.
5. Programs to understand break and continue.
6. Programs to learn writing user-defined functions.
7. Program to learn the concept of recursion.
8. Program to perform searching in an array(linear, binary searches).
9. Program to sort a numerical data set (use bubble sort, selection sort etc.).
10. Program to learn use of multidimensional array. (matrix operations preferred).
11. Programs to understand the use of pointers.
12. Program to practice definition and use of structures.
13. Programs to learn object oriented concepts- definition of a class.
14. Programs to learn the use of constructor.
15. Programs to understand inheritance.
16. Programs demonstrating function overloading and operator overloading.
17. Simple file handling programs – to create, read and write data in file.
18. Simple programming examples in Python using I/O.
19. Programs demonstrating String manipulation using python.
20. Python program using control structures.
21. Python program using loops
22. Python program using collections
23. Python programs using Lists, Tuples, Sets, Dictionaries
24. Python programs using Library Functions & user defined functions.
25. Python programs using classes
26. Programs to learn connecting Python with database.

* * * * * * *

[Scheme 2018]

PGDCA 106 Lab Practice - 2

(PC Skills & Database)

The objectives include familiarizing popular operating systems and application software and practicing SQL. *Programming practice shall include the following topics*:

Part I: (10 Hrs.)

1. Familiarization of Windows operating system and its interface.
2. Familiarization of Ubuntu Linux operating system and its interface.
3. Linux Simple commands and Linux Administration.
4. Familiarization of Word processing packages – MS Word /Libre Office Writer.
5. Familiarization of Presentation Packages – MS Power Point and Libre Office Impress.
6. Familiarization of Image editing softwares- Adobe Photoshop/GIMP etc.
7. Familiarization of MS Excel/Libre Office Calc.
8. Malayalam Computing - enable Malayalam in Ubuntu & Windows. Familiarization of Malayalam Inscript Keyboard Layout Malayalam Typing practice.

Part II: Creating Web Contents (20. Hrs)

1. Practicing basic HTML tags, text tags text styles, paragraph styles, headings, lists
2. Tables in HTML, Frames in HTML, nested frames, Link and Anchor Tags
3. Including graphics, video and sound in web pages, including Java applets
4. Creating HTML document as learned in Module IV of PGD102.
5. Creating CSS contents as learned in Module IV of PGD102.
6. Practice use of Layers & Image Maps.
7. Familiarization of animated Gifs, simple flash animations
8. Familiarising DHTML, VRML.
9. Create your own web pages making use of HTML tags and CSS practiced.
10. Simple Javascript exercises.
11. Writing Javascript functions.
12. Using loops, conditional statements in Javascript.
13. Simple validation experiments with regular expressions.
14. Simple validation experiments without regular expressions
15. Write Javascript programs to draw simple shapes.

Part III: SQL (20. Hrs)

1. Familiarization of RDBMS Environment.
2. Connecting to DBMS. Login and logout. Creating tables
3. Commands to insert records to table
4. SQL commands to describe structure of tables
5. Selecting records from tables, Deleting records.
6. Updating content of tables.
7. Commands to alter tables, drop tables.
8. SQL commands using where clause
9. SQL commands using Having clause
10. SQL commands using Like, in
11. SQL commands using or, and, between, Null
12. SQL commands using Date functions, string functions
13. SQL commands for deleting records, dropping tables.
14. SQL commands using constraints
15. Commands to access records from multiple tables
16. SQL commands for creating views, using join.
17. SQL commands for ordering records, grouping records
18. SQL commands using subqueries.

The practical examination shall include questions from Part II and Part III.

* * * * * * *

[Scheme 2018]

PGDCA201 Advanced Web Technologies

PART-1 ASP.NET

Module 1 : Introduction to .NET- Requirements- .Net Architecture - .Net framework - Common type system - Common language runtime, MSIL. - Components of .Net - Intermediate Language - Just In Time (JIT) Compilation - .NET Base Classes .Introduction – Visual Studio.NET –Components of Visual Studio- Visual Studio.NET Integrated Development Environment - Referencing Components and controls – Projects.

Introduction to web servers, IIS, hosting & virtual hosting - Writing Code - Introduction to ASP.NET - Page HttpRequest Object-HttpResponse Object-Page Directives-@Page directive, @Import directive (5 Hrs)

Module 2 : ASP.NET Web Forms & Controls - Web controls -Link Button Control - Button Control - Label Control - CheckBox Control – RadioButton Control – HyperLink Control - Table Control - Image Control - Panel Control – ImageButton Control - TextBox Control - Simple Web Form Page - List Controls - DropDownList Control,CheckBoxList Control – RadioButtonList Control - Handling Events, AdRotator Control - Calendar Control – Validation controls– RequiredFieldValidator – RangeValidator –CompareValidator – RegularExpressionValidator – CustomValidator

(12 Hrs)

Module 3: Introduction to XML- Introduction to ADO.NET- ADO .Net Architecture – Review of ADO .Net (Introduction to ADO.NET, ConnectionObject, Connection String, CommandObject, DataReaders, DataSets and DataAdapters, DataTable, DataColumn, DataRow, Accessing Data with server explorer ,Working with System.Data.OleDb, Working with SQL.NET, Binding data to controls, Using Stored Procedures)- XML in ADO .Net.- Concept of master page - applying templates. (7Hrs)

PART-2 PHP

Module 4 : (14 Hrs)

Introduction to Server Side Scripting - Installation of LAMP/WAMP - Simple PHP scripts.

Variables and Expressions in PHP - Identifiers , Data Types (integers, floats, strings and booleans)

Creating Variables , Variable assignment,Variable Scope - Namespaces , User-defined Constants.

PHP Operators - Arithmetic Operators , Logical Operators - Relational (Comparison) Operators - The Identity Operator, Bitwise Operators, Other Operators

Conditional Tests - Events and Flow Control in PHP - True and False – If - Comparing Strings and Numbers - else and elseif - The ? : Operator - Comparing for equality or identity - The switch / case statement - Loops - while, do ... while, for - break and continue - exit

Taking User Input from Forms - Review of HTML forms. - Creating forms with PHP GET and POST Methods - Security Protection from Cross Site Scripting - Obtaining data from forms through variables - Using hidden fields - Secure Web Forms(HTTPS and SSL) - Calling pages from themselves using forms

PHP Functions - Purpose of functions - Built-in functions - Declaring functions - Functions Arguments -Default Arguments - Return - Returning References - Variable Scope (global vs local) - Static variables - Recursion - include and require - Variable function calls.

Storing Data in Arrays using PHP - Indexing arrays, Using foreach - Add/remove elements from arrays -One-dimensional arrays - Multi-dimensional arrays - References to array elements.
String Manipulation in PHP - substring and [] operators - Case conversion, trimming - Formatting output data - Regular expressions.

Module 5 : (12Hrs)
Introduction to mysql - MySQL command Interpreter - Creating a database - Modifying a database - Administering a database.
Developing an application using PHP and MySQL - MySQL functions in PHP - Querying a database - UserDriven Querying - Inserts, Updates, and Deletes - User Authentication and Management.
Concept Sessions and Cookies in PHP - Understanding PHP session variables - Create a PHP session variable - Register and modify PHP session variables - Cookie properties - Setting a cookie in PHP - Retrieving PHP – cookies - Expiring/Deleting PHP cookies - Drawbacks to cookies.
PHP objects - Object basics - Classes in PHP - constructors and destructors -
Inheritance - Accessing instance data and functions - __sleep and __wakeup - Class constants.

Text: 1. ASP .Net Compete Reference by Mathew MacDonald – TMH
2. Beginning Php 5 - Dave W. Mercer, Allan Kent, Steven D. - Wiley India

Ref: 1. ASP .Net Unleashed, VB .Net programming – Wrox publication
2. Web References Learning Php 5 - DavidSklar - Shroff/o'reilly

* * * * * * *

[Scheme 2018]

PGDCA202 Programming in Java & Android

Module 1 (6 Hrs)

Review of Object oriented concepts – class, object, encapsulation, data hiding, abstraction, Polymorphism, inheritance.
Features of Java- Platform independence, garbage collection, Byte code, virtual machine, web applications.
Components of JDK environment – javac, java, jar, javadoc etc.
Programming concepts- Identifiers, Data types, operators, expressions
Java coding conventions, input and output operations, programs using classes and objects, scope and visibility modes, static members, wrapper classes, command line arguments , instance and reference of objects. Constructors, finalize method, overloading.
Control structures- if, if..else,switch, goto, looping statements, break, continue, labeled break, labeled continue.

Module 2 (8 Hrs)

Inheritance- extends keyword, subclass and superclass, use of this & super - method overriding - final keyword – Interfaces- implements keyword
Brief overview of Java API packages – lang, util, io, net, applet, awt, sql
Arrays, Strings, String handling functions.
Packages- package, import statements, user defined packages
Exception handling – try, catch, finally, throw and throws keywords, built in exceptions
Multi threading - definition, Thread Life cycle, priority of threads, synchronized keyword, inter thread Communication

Module 3 (16 Hrs)

Applets- Java Graphic fundamentals, applet life cycle, Drawing Text, Lines and Shapes (rectangles, ovals, Arcs, Polygons)- Colors, Fonts, Displaying images, Reading Parameters.
Creating user interface components using Javax.swing package(JButton, JTextField, JLabel, JCheckbox, JRadioButton, JComboBox, JList, JPanel, JMenu, JMenuBar, JMenuItem, JOptionPane, JFrame, JApplet). Layout Managers- FlowLayout, GridLayout, CardLayout, Border Layouts.
Event driven programming -ActionEvent, ActionListener, ItemEvent, ItemListener,MouseEvent, MouseListener, MouseMotionListener, KeyEventKeyListener, WindowEvent, WindowListner - Creating stand alone GUI applications. Java.util package- List& String Tokenizer.
Java I/O package- File, FileInputStream, FileOutputStream, FileReader, FileWriter, BufferedReader, PrintWriter, DataInputStrem, DataOutputStream, Stream Tokenizer, Scanner.
Database programming - JDBC - ODBC - Components of JDBC API-Connecting to a Database – Statements and Prepared Statements - Executing queries - updating database

Module 4 Android Programming (20 hrs)

Introducing the android computing platform, History of android, android software stack, Developing end user application using androidSDK, android java packages, Setting up the development environment, Installing android development tools (ADT), Fundamental components-, Android virtual devices Dalvik Virtual Machine & .apk file extension, Running on real device, Structure of android application, Application life cycle
Basic Building blocks - Activities, Services, Broadcast Receivers & Content providers
UI Components - Views & notifications- Components for communication -Intents Android API levels (versions & version names) AndroidManifest.xml Resources & R.java , Assets, Layouts & Drawable Resources Activities and Activity lifecycle- First sample Application- Basic UI design
Form widgets, Text Fields, Layouts-Alert Dialogs & Toast.

Text: Thinking in Java - Bruce Eckel.

Ref :
1. Programming with Java - Balaguruswamy
2. Java unleashed - Second Edition - Michael Morrison
3. Java Complete Reference - Herbert Schildt
4. Pro Android 4, Satya Komatineni& Dave MacLean, Apress.

* * * * * * *

[Scheme 2018]

PGDCA203 Advances in Information Technology

Module 1: E-Commerce ,M-Commerce & Digital money (14 Hours)

E-Commerce & Digital payments: Business Models - B2B, B2C, B2G, C2C, P2P& B2E – Requirements of e-commerce website – Modes of delivery- M-Commerce- purchases on mobile web and apps,mobile payments,mobile money transfers and m-banking-and mobile financial services.

Digital payments: USSD, NUUP, AEPS, UPI, Debit card, Credit card, Pre-paid cards, Internet Banking, Mobile banking, e-wallets – POS Terminals, M-POS, V-POS.- Concept of Crypto Currency.

Module 2: Internet Security (12 Hours)

Internet Security - Security need,Security threats, Security methods- Authorisation- Encryption basics-

Digital Security – Security threats in Internet & e-Commerce, Virus & Trojan horses, Phishing, Hacking, Spyware – Security Measures, Antivirus, Firewall, Encryption, Bio-metrics –Precautions while using Internet & e-payments - IT Act, Major Provisions of IT Act.

Module 3:E-Governance and Technical Writing (12 Hours)

Defining E-Governance- Types of interactions in E-Governances - G2G,G2C,G2B,G2E - Benefits - examples- FRIENDS, AKSHAYA, e-District etc.

Need for Technical writing, Role of a Technical Writer,Technical Writing Process-

Document development process - Estimating Technical Documentation , Documentation Planning, Selection of Tools, Templates and Page design , Content Development , Formatting and pagination , Content Publishing - Technical Writing Software Tools.

Module 4: Content Management system. (12 Hours)

Content Management Systems : Content – defining data, information and content, content format, structure, functionality is content, what is content management – Understanding content management, introducing the major parts of a content Management System.

Case study: Drupal/Wordpress/Joomla.

Reference:

1. PTSJ Joseph, E-Commerce: An Indian perspective, PHI
2. Taxman-Digital banking,Indian Institute of Banking &Finance

* * * * * * *

[Scheme 2018]

PGDCA204 Lab Practice - 4

(ASP.NET, PHP)

Part A - ASP.Net (26 hrs)

Lab Activities

1. Familiarization of Visual Studio IDE
2. Introduction to Web Forms
3. Demonstration of Web Hosting & Virtual Hosting
4. Simple web applications using standard webcontrols
5. Web application for handling form events
6. Web applications for handling keyboard and mouse events
7. using calender control
8. Programs using adrotator control
9. Programs using different validation controls
10. Programs to demonstrate trace.
11. web applications to display records in a table.
12. web applications to display records in a data grid
13. displaying XML data in form
14. web applications to Insert, Delete, Search, Update records in a table.
15. Binding data to grid
16. Binding data to controls like combo box.
17. Programs using stored procedures
18. Programs using master page
19. Programs using templates

Sample Programs

1. Develop a Web application to find the biggest of 2 numbers entered in a Textbox.
2. String processing: Accept two strings and perform the following:
   (1) String concatenation
   (2) Case conversion
   (3) Determine the length
   (4) Reverse the string
3. Create a registration form and include validation control.
   (Use the Ispostback property)
4. Create a student information form and insert data into the database and display the result in a data grid.
5. Assume that a student database is given. Perform the following operation:
   (1) Search (2) Update (3) Delete
6. XML data binding
7. Implement a stored procedure
8. Implement AdRotator control
9. Design an to ASP.NET application to prepare student mark sheet using any database.
10. Design an to ASP.NET application to prepare employee pay slip using any database.
11. Develop a web application to display records using Data Grid view.
12. Programs for demonstrating event handling.
13. Program using menus.
14. data binding

Part B - PHP (24 hrs)

Installation of lamp server
Installation of wamp server
Simple web applicarions using PHP username, password authentication, table look up, mark list etc
Webforms . secure webforms
Sessions Cookies Session tracking
PHP and mysql- PHPMYadmin.
Building data base driven applications from the case study.

Recommended excercises (Any four)

Marklist
user authentication with password using Mysql
user authentication with password using Mysql and sessions and cookies
Students Database
Digital Library
College Admission

* * * * * * *

[Scheme 2018]

PGDCA205 Lab Practice - 5

(Java & Android)

Note : Lab record should contain minimum 20 programs (Minimum 4 programs from Android)

Part 1. JAVA (30 hrs)

Simple programs using control statements
Programs using class, object, constructors and destructor concepts
Simple programs using command line arguments

Programs to demonstrate inheritance, overloading and Overriding
Programs using Package and Interface concepts
Programs handling arrays, sorting, searching, managing matrices
Programs for string processing using string functions
Programs to demonstrate the working of language defined exceptions

Simple applets
Applets to draw Lines and Shapes (rectangles, ovals, Arcs, Polygons)
Applets that get parameters from html pages
Programs to demonstrate the behaviour of JButton, JTextField, JLabel,JCheckbox, JRadioButton, JComboBox, JList, JPanel, JOptionPane, JFrame, JApplet
Programs to display images and text in various fonts
Sample programs using JMenu, JMenuBar, JMenuItem, to create menus
Sample programs using various layout managers
Sample programs using various listeners
Sample programs using multi threading.

Programs to demonstrate the behaviour of List, String Tokenizer.
Programs using various streams
Programs using various file streams
Simple programs for various database operations (database querying, updating)

Part 2. Android (20 hrs)

Setting up development environment- Familiarisation of Eclipse/Android studio

1. Hello World Program (Write a program to Toast Hello World)
2. Addition of two Numbers (Write a program to add two numbers)
3. Date and Time Dialog box (Write a program to display date and time using dialog box)
4. Alert Box (Write a program to Display an alert box with OK and Cancel)
5. Radio Button (Write a Program to select gender using radio button)
6. Spinner (Write a Program to spin the four items)
7. Timer Program (Write a Program to display Stop watch)
8. Check box (Write a Program to check the items listed)
9. Program to toast a message.

* * * * * * *

[Scheme 2018]

PGDCA206 Project Work

(50 Hrs)

1. Aim: To expose student to industry-standard project practices, through a real-life project work under time and deliverable constraints.

2. Project Guidelines

The minimal phases for the project are: Project search, finalization and allocation, Investigation of system requirements, Data and Process Modeling, System Design, Program design, Program coding and unit testing, System integration, System implementation .

3.1 Project Planning: The topic should be chosen in Semester 1 itself , though on exceptional cases, for valid reasons, the project guide may waive this condition. Related reading, training and discussions should start from semester 1 itself. Project work should be competed within 60 Hrs of Lab duration.

3.2 Selection of project work: The project can be any type of the following.

(a) Developing solution for a real-life problem : Examples are Accounting Software Package for a particular organization, Computerisation of administrative functions of an organization, Web Based Commerce, etc.

(b) Innovative Product development: These are projects where a clear-cut requirement for developing a computer based solution may not be existing, but a possible utility for the same is conceived by the proposer. An example is a Malayalam Language Editor with Spell Checker, Simulation Software for eLearning, Digital Water Marking Software, ..

(c) Research level project: These are projects which involve research and development and may not be as structured and clear cut as in the above case. Examples are Malayalam Character Recognition, Neural Net Based Speech Recogniser, Biometric Systems, Machine Translation System etc. The methodology and reporting of such projects could be markedly different from type (a) and is left to the proposer/external supervisor of the projects

3.2 Selection of Team: To meet the stated objectives,it is imperative thatMajor Project isdone through a team effort**.** Though it would be ideal to select the team members at random (drawing lots) and this should be strongly recommended, due to practical considerations, students may also be given the choice of forming themselves into teams with 3 to 5 members (teams less than 3 members may be permitted in exceptional cases, for valid reasons). A gender mix should also be strongly suggested. A team leader shall be elected .Teams shall maintain team meeting minutes and ensure that every team member has tasks assigned in writing. Team meeting minutes shall form a part of the Project Report. Even if students are doing projects as groups, each one must independently take up different modules of the work and must submit the reports also independently (though, in such cases, some common materials is permissible). Evaluation will also be done independently.

3.3 Selection of Tools: No restrictions shall be placed on the students in the choice of platforms/tools/languages to be utilized for their project work, though open source is strongly recommended, wherever possible. No value shall be placed on the use of tools in the evaluation of the project.

Students may also choose to do project in the college/institute (or partially in the college/institute and partially in an external organization), especially product-based work, but in such cases the supervisors must ensure that the industry practices are followed .

3.4 Documentation:

One copy should be maintained by each student and a copy from the group to the Institution library. The format for preparation of the project report is standardized (students need not consult earlier project reports). The following are the major guidelines: The final outer dimensions of the report shall be 21 cm X 30 cm. The colour of the flap cover shall be light green. Only hard binding should be done, with title of the thesis and the words "<BRIEF TITLE> PGDCA Project Report 201…" displayed on the spine in 20 point, Bold, Times New Roman, as in example below. In case the title is too long, a shorter version of it may be used (Like "Image Pro" instead of "Image Pro – An Interactive Image Processing package").

- The text of the report should be set in 12 pt, Times New Roman/Calibri/Cambria, Single Spaced.
- Headings should be set as follows: CHAPTER HEADINGS 20 pt, Times New Roman, Bold, All Caps, Centered.

1. SECTION HEADINGS 12 pt, Times New Roman, Bold, All Caps, Left Adjusted.

1.1 Section Sub-headings 12 pt, Times New Roman, Bold, Left Adjusted.
Titles of Figures, Tables etc are done in 12 point, times New Roman, Italics, Centered.
Some general guidelines on documentation stylistics are:

- Double quotes and single quotes (“”, “) should be used only when essential. In most cases words put in quotes are better highlighted by setting them in italics
- Page numbers shall be set at right hand top corner, paragraph indent shall be set as 3.
- Only single space need be left above a section or sub-section heading and no space may be left after them.
- Certificate should be in the format: “Certified that this report titled....................... is a bonafide record of the project work done by Sri/Kum....................... under our supervision and guidance, towards partial fulfillment of the requirements for the award of the Post Graduate Diploma in Computer Application of IHRD” with dated signatures of Internal Guide and Head of Institution.
- If the project is done in an external organization, another certificates on the letterhead of the organization is required: “Certified that his report titled.............................. is a bonafide record of the project work done by Sri/Kum............................ under any supervision and guidance, at the ....Department of.................... (Organization) towards partial fulfillment of the requirements for the award the Post Graduate Diploma in Computer Application of IHRD “
- Space for signature of Internal& External Examiners shall be provided in the facing sheet or index page.
- References shall be IEEE format (see any IEEE magazine or transaction). Take care in use of italics and punctuation. While doing the project, keep note of all books you refer, in the correct format, and include them in alphabetical order in your reference list. Eg: A book is cited as: Kartalopoulos, S V Understanding Neural Networks and Fuzzy Logic, BPB Publishers, 1996, pp. 21-27. (pp.21-27 indicates that pages 21-27 have been referred. If the whole book is being referred, this may be omitted. If a single page is referred, say 7, it may be cited as p.7

<PROJECT TITLE>

<STUDENT’S NAME>
<COLLEGE/INSTITUTION NAME>

PROJECT REPORTSUBMITTED IN PARTIAL FULFILMENT
OF THE REQUIREMENTS FOR THE AWARD OF
POST GRADUATE DIPLOMA IN COMPUTER APPLICATION OF IHRD
2018

Drafts should be read, modified, spell checked and grammar checked at least twice during the course of the project and before a final printout is taken, the same may be got approved from the internal guide. The students should send two interim reports to internal guides. This will also help the students in their report writing.

Regarding the body of the report, as an indicative example, the following is given (though students should not attempt to fit every kind of project report into this format):

- Organizational overview (of the client organization, where applicable)
- Description of the present system
- Limitations of the present system
- The Proposed system- Its advantages and features
- Top level DFD of the proposed system with at least one additional level of expansion
- System flowchart
- Menu Tree
- Program List
- Files or tables (for DBMS projects) list. Class names to be entered for each file in OO systems.
- List of fields or attributes (for DBMS projects) in each file or table.

- Program – File table that shows the files/tables used by each program and the files are read, written to, updated, queried or reports were produced from them.
- Reports List with column headings and summary information for each report.
- System Coding and variable/file/table naming conventions
- Screen layouts for each data entry screen.
- Report formats for each report.

Program documentation is suggested on the following lines:

- Program id
- Program function explanation
- Data entry screen (reproduced from system documentation).
- Report layout (reproduced from system documentations)
- Program level pseudocode or flowchart.
- Decision tables, decision trees, with English explanation where necessary.
- Program listing
- Test data & Test results.

3.5 Methodology:

Wherever applicable, object oriented approach should be used for software development. The project report should generally contain details of the following steps (though students should not attempt to fit every kind of project into this format):

(a) Analysis
- Study of existing systems and its drawbacks (general)
- Understanding the functionalities of the system (detailed)
- Preparation of requirement
- Conduct of Feasibility study
- Identification of relevant Objects
- Abstraction of each object (attributed and methods)
- Relationship between objects

(b) Design
- Design of each subsystems
- Design of each classes
- Design of communications between objects
- Design of Algorithms for problem solving
- User interface Design
- Any other steps if necessary

(c) Coding and Implementation
(d) Testing
(e) Security, Backup and Recovery Mechanisms
(f) On line help and User Manuals
(g) Upgradability Possibilities

3.6 Project IPR & Utilisation: The intellectual property rights in all project work done by the students shall vest with IHRD, except in cases where some external organizations seek undertaking from students to concede IPR in all work done in their organization or under their guidance. Where possible, students should attempt to obtain at least a joint IPR for IHRD. In cases where project works are of public utility, students shall be asked to publish their work including source code and documentation, in so far as their rights are clear.

4. References

4.1 Core References

- S A Kelkar, Software Project Management, Prentice Hall of India
- W Alan Randolph, Barry Z. Posner, Effective project planning and management, Prentice Hall of India, ISBN 0-87692-776-2

4.2 Additional References

- Greg Mandanis, Software Project Management Kit for Dummies, IDG Books, ISBN 81-265-0100-6
- Joel Henry, Software Project management, ISBN 0-201-75865-2

* * * * * * *

[Scheme 2018]

POST GRADUATE DIPLOMA IN COMPUTER APPLICATIONS

1. Question paper pattern

Duration of Exam. : 3 Hrs. Maximum marks : 100

Part - A Multiple choice / fill in the blanks type questions
Part - B Short Answer type Questions with answer size up to 1 page per question.
Part - C Descriptive type Questions with answer size up to 2 to 3 pages per question.

Marks Distribution

Part	No. of questions.	Need to be answered	Marks/Question	Total
A	20	20	1	20
B	10	8	5	40
C	6	4	10	40
Total		32	…	100

Guidelines for question paper setters:

1. Each part should cover questions from each module in the syllabus.

2. The level of difficulty shall be as follows

i)	Easy Questions	: 30% -40%
ii)	Intermediate level to difficult	: 30% -40%
iii)	Difficult questions	: 20% -30%

3. The question paper setters must prepare and submit the question papers as per the following guidelines.
   a. Question paper must be designed and prepared to fit in an A4 size paper with one inch margin on all four sides.
   b. Prepare the Question in MS-Word/Open office-Write document format. Use only “TimesNewRoman” font with size 10. Align text to both left and right margins.
   c. Please leave 5 cm. free area at the top of the front page of each question paper to place examination details/Question paper header by the examination department.
   d. Avoid placing 1 or 2 questions in the last part in a fresh page, unless it is absolutely necessary. In such case, try to accommodate above questions in the previous page(s) by adjusting top/bottom margins and line spacing, if possible. This will reduce printing expenses.
   e. Specify marks for each question/part clearly.
   f. Clearly specify the number of questions to be answered for each Part.
   g. Confirm that no questions in part B is repeated in Part C also.
   h. Avoid repeating questions in Part C from the immediate previous examination.
   i. Key for evaluation must be prepared and enclosed in a separate cover and should be submitted along with the question paper set. Key for evaluation must specify evaluation guidelines for each part in the question paper, otherwise the key prepared will be treated as incomplete.
   j. Submit Question paper in Laser print out form only. Hand written and printed in poor quality printers is not acceptable.

****

[Scheme 2018]

POST GRADUATE DIPLOMA IN COMPUTER APPLICATIONS

2. Scheme for Continuous Evaluation.

1. For Theory Papers :	Weightage
a). Average of minimum Two test papers	: 30 %
b). Average of minimum Two Assignments	: 30%
c). Score for Seminar	: 20%
d). Score for Class Attendance.	: 10%
e). Overall performance in the class.	: 10%

2. For Practical Papers :	Weightage
a). Average of minimum Two Lab tests	: 30 %
b). Average of minimum Two Lab Assignments	: 30%
c). Maintenance of Lab record	: 20%
d). Score for Lab Attendance.	: 10%
e). Overall performance in the Lab.	: 10%

3. Teachers shall submit Mark list for Continuous Evaluation to the Head of Institution in the following format.

Subject:

Sl no.	Regno.	Name	a.Test	b.Assignment	c.Seminar	d.Attendance	e.Performance	Total

4. Head of Institution/Co-ordinator shall forward Continuous evaluation marks to the Examination Department in the following format only.

Centre code : Centre name:

Sl no.	Regno.	Name	PGDCA101 50	PGDCA102 50	PGDCA103 50	PGDCA104 50	PGDCA105 50	PGDCA106 50

5. Continuous evaluation(CE) marks must be published in the notice board at least one week before the commencement of theory examinations after getting approval from the Head of Institution/Co-ordinator.

****

Thiruvananthapuram
February 21, 2018

Sd/-
Director