Source Book
of
Certificate Course in iOS Programming

Table of Contents

Source book for C-DAC's Certificate Course in iPhone Programming

1. **Course Objective:** The objective of this course is to provide the student with an expertise in iOS Programming. After doing the course the student will be able to design, develop and maintain iOS applications.

2. **Eligibility Criteria:** Any Engineering /Science graduate with mathematics up to Senior Secondary or 10+2 level

3. **Prerequisite:** Sound knowledge of Computing Fundamentals and Fundamentals of Programming.

4. **Teaching Schema: (Tabular format)**

Sl. No.	Module Name	Hours
1	Fundamentals of Computer & OS Concepts	20
2	Objective C and OOPs concepts	60
3	Software Development Life Cycle	16
4	Database Technology	20
5	Mobile and Wireless Technologies	20
6	iOS Programming	84
7	Management Development Program	60
8	Project	40
	Total	**320**

5. **Suggested Schedule**

Week	Teaching Sessions & Academic Activity
1	Fundamentals of Computer & OS Concepts (20/20 Hours) **and** Objective C and OOPs concepts (10/60 Hours)
2	Objective C and OOPs concepts (30/60 Hours)
3	Objective C and OOPs concepts (20/60 Hours) **and** Software Development Life Cycle (10/16 Hours
4	Software Development Life Cycle (6/16 Hours) **and** Database Technology (20/20 Hours) **and** Mobile and Wireless Technologies (4/20 Hours)
5	Mobile and Wireless Technologies (16/20 Hours) **and** iOS Programming (14/84 Hours)
6	iOS Programming (30/84 Hours)
7	iOS Programming (30/84 Hours)
8	iOS Programming (10/84 Hours) **and** Management Development Program (20/60)
9	Management Development Program (30/60)
10	Management Development Program (10/60 Hours) **and** Project (20/40 Hours)
11	Project (20/40 Hours) **and** Exam Break (2 days)
12	1st Day – Exam, Two Days – Project Evaluation, 5th Day – Re-exam

Note: Course Delivery is 6 hours per day, 5 days per week (30 hours per week)

6. Session wise Breakup

Note: Each single session is of two hours duration for all subjects mentioned below.

Fundamentals of Computer & OS Concepts

Theory 10 hours + Lab 10 hours

Session 1:

- Computer Fundamental
- Types of computer
- Primary and Secondary storage
- Input-output devices
- Disks, tapes and CD-ROMs.
- VDUs, printers and other output devices
- Introduction to windows operating systems
- The desktop, The window, application window, document window, Dialog Window
- The Icons, Explore Your Computer, The Start Button and Taskbar.
- My Computer, Windows Explorer, Starting and Closing Programs,
- Installing Operating System
- Performing a New Installation for Windows
- Installing a Software other than OS
- Setting up a printer
- Uninstalling software

Session 2:

- Overview of Operating systems
- Types of Operating System
- What is process and thread
- Process management

Session 3:

- Process Scheduling
- CPU Scheduling
- Preemptive vs Non-Preemptive

Session 4:

- Deadlocks
- Memory Management
- Memory and I/O Maps

Session 5

- File Management
- I/O Management
- IPC

Lab Assignments

1. Installing OS and Install, Update and maintain Packages in OS
2. Getting Acquainted with the Linux Environment and Use various commands in Linux system (ls, cp, mv, lpr, sort, grep, cat, tac, more, head, tail, man, whatis, whereis, locate, find, diff, file, rm, mkdir, rmdir, cd, pwd, ln and ln –s, gzip and

gunzip, zip and unzip, tar an its variants, zcat, cal, bc and bc –l, banner date, time, wc, touch, echo, who, finger, w, whoami, who am i, alias, unalias, touch, push, pop, jobs, ps, etc.)

3. Getting confidence of the uses of Windows 7 and Creating different documents using windows application related to Programming
4. Problems solving on CPU scheduling
5. Implement to show the Synchronization of Threads using Semaphores.

Objective C and OOPs concepts

Theory 20 hours + Lab 20 hours

Session 1:

- Introduction to Objective-C programming
- Primitive Data Types and Operators
- Expressions (Arithmetic)
- Operator precedence
- Compiling and running programs

Session 2:

- Arrays and Structures
- Flow Control Statements
- Looping and branching
- Break & Continue
- Nested if statements
- Switch statements
- Discussion on Classes
- Discussion on Objects
- Discussion on Messaging
- Discussion on Instances
- Accessing Instance Variables and Data Encapsulation
- Working with Fractions

Session 3:

- Calculator class and other built in classes
- Inheritance
- Interface and implementation of files
- Extending Class definitions and the Interface file

Session 4:

- Polymorphism
- Dynamic Typing and Dynamic Binding

Session 5:

- Protocols and Categories
- Delegations
- Informal protocol
- Composite objects

Session 6:

- Memory Management and Properties
- Garbage Collections
- Object References and Autorelease pool
- Strong variable and weak variables

Session 7:

- Introduction to Foundation Framework Classes
- Property Lists, NSCopy, and Archiving
- Selectors and Targets

Session 8:

- Difference between C and C++
- Introduction to C++: Identifier, Keywords, Constants,
- Operators: Arithmetic, relational, logical, conditional and assignment. Sizeof operator, Operator precedence and associativity.
- Type conversion, Variable declaration, expressions, statements, manipulators
- Input and output statements, stream I/O, Conditional and Iterative statements, breaking control statements.
- OOP Concepts
- Class and Objects
- Executing sample C++ programs

Session 9:

- Static Member
- Inline Function
- Access specifiers
- String & Streams
- Constructors and Destructors
- Properties
- Types of constructors & destructors
- Inheritance
- Types of inheritance

Session 10:

- Friend functions
- Friend Classes
- Polymorphism
- Types of polymorphism
- Overloading functions
- Overloading Operators
- Copy Constructors
- Run Time Polymorphism
- Virtual Functions

Lab Assignments:

1. Write a program in Objective-C to display the following text:
   "In objective-C, lowercase letters are significant.
   main is where program execution begins.
   Open and Closed braces enclose program statements in a routine.
   All program statements must be terminated by a semicolon."
2. Write a Objective-C program that subtracts the value of 15 from 87 and displays the result, together with an appropriate message.
3. You have a boat and a motorcycle in addition to a car. List the actions you would perform with each of these. Observe do you have any overlap between these actions. If yes name them.
4. Define a class called XYPoint that will hold a Cartesian coordinate (x, y), where x and y are integers. Define methods to individually set the x and y coordinate of a

new point and retrieve their values. Write an Objective-C program to implement your new class and test it.

5. Write a Objective-C program that converts 27^{O} from degrees Fhrenheit (F) to degrees Celsius (C) using the following formula:
   C=(F-32)/1.8
6. Write a program in Objective-C to evaluate the polynomial shown below: (remember to use exponential format to display the result)
   $3x^3 – 5x^2 + 6$ for x = 2.55
   $(3.31 \times 10^{-8} + 2.01 \times 10^{-7}) / (7.16 \times 10^{-6} + 2.01 \times 10^{-8})$
7. Write Objective-C program that define a new class called complex. Do addition of two complex numbers in your program.
8. Write a program in Objective-C to generate and display a table of n and n^2, for integer values of n ranging from 1 through 10. Be sure to print the appropriate column headings.
9. Write a program in Objective-C that calculates the sum of the digits of an integer. For example, the sum of digits of the number 2155 is 2+1+5+5=13. Your program should accept any arbitrary integer from user.
10. Write a program in Objective-C that takes an integer keyed in from the terminal and extracts and displays each digit of the integer in English. For example if user types in 932, your program should display the following:
    Nine
    Three
    Two
11. iOS uses a structure called CGRect for working with rectangles. Modify the XYPoint and Rectangle classes to deal with floating-point numbers. The rectangle's width, height, area and perimeter should all work with floating-point numbers as well.
12. Using typedef and enumerated data types, define a type called Day with the possible values Sunday, Monday, Tuesday, Wednesday, Thursday, Friday and Saturday.
13. Define a macro clled MIN that gives the minimum of two values. Then write a program in Objective-C to test the macro definition.
14. Define a macro called MAX3 that gives the maximum of three values. Write a program in Objective-C to test the definition.
15. Write a function that calculates the average of an array of 10 floating-point values and returns the result.
16. Write a function to add all the Fractions passed to it in an array and to return the result as Fraction.
17. Look up the NSDate class in your documentation. Then adda new category to NSDate called ElapsedDays. In that new category, add a method based on the following method declarations:
    – (unsigned long) elapsedDays: (NSDate *) theDate;
18. Add new fields of your choice to the AddressCard class. Separate the name field into first and last name fields and adding address (with separate state, city, postal code and country fields). Write appropriate setter and getter methods and ensure that the print and list methods properly display the fields.
19. Write a program in Objective-C that takes an NSArray of NSNumber objects (each NSNumber represents integer) and produces a frequesncy chart that lists each integer and how many times it occurs in the array. Use an NSCountedSet object to construct your frequency.

20. Open a file and read its contents 128 bytes at a time and write it to the terminal. Use NSFileHandle's fileHandleWithStandardOutput method to obtain a handle for the terminal's output.
21. A dictionary is stored as a property list at this URL: http://bit.ly/aycNwd. Write a program in Objective-C to read the contents of this dictionary and display its contents. What data does the dictionary contains?
22. Create an NSDictionary object and fill it with some key/object pairs. Then make both mutable and immutable copies. Are these deep copies or shallow copies that are made? Verify your answer.
23. Write a program in Objective-C to read in an archived AddressBook and look up an entry based on a name supplied on the command line like below:
    $ lookup soumya
24. Design a calculator application with good UI and having user friendly functionalities.
25. Write a C++ program which accept two numbers and print their sum.
26. Write a C++ program which accept principle, rate and time from user and print the simple interest.
27. Write a C++program to check whether the given number is even or odd (using ? : ternary operator)
28. Write a Student class and use it in your program. Store the data of ten students and display the sorted data according to their roll numbers, date of births, and total marks.
29. Write the definition for a class called **time** that has hours and minutes as integer. The class has the following member functions:
    void settime(int, int) to set the specified value in object void showtime() to display time object time sum(time) to sum two time object and return time.
    a. Write the definitions for each of the above member functions.
    b. Write main function to create three time objects. Set the value in two objects and call sum() to calculate sum and assign it in third object. Display all time objects.
30. Write a C++ program using basic concept of objects and classes to check whether given number is prime or not.
31. Using virtual and pure virtual functions implement hierarchy of computer printers.
32. Design a hierarchy of computer printers. Use multiple inheritances in your hierarchy. Also use friend functions and classes in your program.
33. Write Date and Time classes that allows you to add, subtract, read and print simple dates in dd/mm/yyyy and time in hh:mm:ss forms. Use function overloading in your program.
34. Write C++ programs to overload =, ==, +, ++, --, <<, >> and [ ] operators
35. Write a template class for sorting method. Using this class, write a test program for sorting using different data types. Also implement exception handling.
36. Create a text file using any simple editor. Write a program that will reverse each line in the input file and store it in another file.
37. Create a C++ class named Grade that has the same functionality as the old struct Grade and associated functions in the original C file. When you are finished, your C++ file should only have one function definition outside of the class: main().
38. Write a menu driven program in C++, which consist followings:
    - To read and display the transpose of MxN matrix
    - To sum of two MxN matrix

- To product of two MxN matrix

Software Development Life Cycle

Theory 16 hours

Session 1:

- Introduction to Software and Software Engineering
- Software Process
- SDLC and Process Models
- Agile Development model
- Case study on Agile

Session 2:

- Requirement Engineering
- Requirement analysis
- Use case approach
  - Use cases & usage scenarios
  - Identifying use cases
  - Use cases & functional requirements
  - Benefits of Use cases

Session 3:

- Design concepts
- Component Definitions
- Class-based Component Design
- Class-based Component Design Principles
- Component-Level Design Guidelines
- Components
- Function-Oriented Software Design

Session 4:

- Introduction to Coding Standards
- Best Coding Practices
- Commenting and Code understanding
- Documents on Coding

Session 5:

- Testing Strategies and Tactics
- Writing Test Cases
- Different Testing methods
- Software implementation
- Phased wise deployment
- Software maintenance
- SLA on maintenance

Session 6:

- Software Project Management
- Project Management Definition

- Distinctive characteristics of Software
- Scope, Organizing, Planning,
- Scheduling, Graphical Schedule representations
- Activity Organization, Milestones, Deliverables

Session 7:

- Task Dependencies,
- Staffing, Communication
- Project Risk identification, analysis, planning monitoring
- Monitoring, Reviews,
- Control, Reporting
- Project Closure

Session 8:

- Object-Oriented Software Development
- Software Reuse
- Emerging Trends and their overview
- Case studies

CASE STUDY- I

Communities as well as non- profit and charitable organizations promoting reuse as an alternative to discarding unwanted items. It also covers market place functionality. User can buy and sell various products. Companies can also post details of the stuff they want to get rid of in bulk. This is a secure site where SSL has been used. Site traffic analyzer is also available to the site admin. It also has a blog associated with it where the admin can post case studies of the user experience.

The admin has the ability to create admin roles of various levels, view monthly transactions, create communities, hauling company details, customize email template and specify content, also give the feature in which to the admin where any corporate can contact the admin and they can provide details of the various goods that they want to get rid of in bulk.

For analysis of project please include followings documents:

1. SRS (software requirement specification)
2. SDLC (software development life cycle)
3. Database & technologies required
4. Project Mangement
5. Quality Management
6. Cost evaluation
7. Time requirement for the completion
8. Training ,Maintance & Documentation

Design a system which will meet all the above needs of the organization. The system should be flexible and should provide growth in the organizational activities.

CASE STUDY II

Objective

The objective of the project is to develop a software to assist ABC foundation in making decisions regarding home loans .The product will allow the client to add modify and delete information regarding the foundation investment, operating expenses and individual information. The product will perform the required calculations in the areas and produce the listing investment.

Assumptions & constraints

1. The deadline must be decided and met
2. validation and fulfillment of all documents for processing
3. the product must be reliable
4. The architecture must be open so that additional functionality may be added later
5. The product must be user friendly.

For analysis of project please include followings documents:

1. SRS (software requirement specification)
2. SDLC (software development life cycle)
3. Database & technologies required
4. Project Management
5. Quality Management
6. Cost evaluation
7. Time requirement for the completion
8. Training , Maintance & Documentation

Design a system which will meet all the above needs of the organization. The system should be flexible and should provide growth in the organizational activities

Database Technology

Theory 10 hours + Lab 10 hours

Session 1:

- Introduction to DBMS – What is DBMS, Its need
- Areas where DBMS are used
- Types of DBMS: Introduction to Hierarchical Model, Network and Relational Models
- Data models (conceptual physical and logical)
- Data Integrity & integrity rules

Session 2:

- Introduction to Oracle
- SQL* Plus
- DDL Commands
- DML & DCL Commands
- Inbuilt Functions
- Grouping Things Together (Group By, Having Clause)
- Advance Subqueries (Correlated Sub query, Outer Joins)

Session 3:

- Set Operators (UNION, UNION ALL, INTERSECT, MINUS)
- Types of Views
- Creating Complex View
- Using Temporary Tables

Session 4:

- Introduction to PL/SQL
- PL/SQL Programming style
- PL/SQL Program blocks
- Cursors

Session 5:

- Procedures & Functions

- Triggers

Lab Assignments:

1. Write a program that computes the perimeter and the area of a rectangle. Define your own values for the length and width. (Assuming that L and W are the length and width of the rectangle, Perimeter = 2*(L+W) and Area = L*W. Display the output on the screen using dbms_output.put_line.
2. Write a program that declares an integer variable called num, assigns a value to it, and computes and inserts into the tempp table the value of the variable itself, its square, and its cube.
3. Convert a temperature in Fahrenheit (F) to its equivalent in Celsius (C) and vice versa. The required formulae are:-
   C= (F-32)*5/9
   F= 9/5*C + 32
   Display the output on the screen using dbms_output.put_line. Data has to be input by the user.
4. Convert a number of inches into yards, feet, and inches. For example, 124 inches equals 3 yards, 1 foot, and 4 inches. Display the output on the screen using dbms_output.put_line. Data has to be input by the user.
5. Write a program that enables a user to input an integer. The program should then state whether the integer is evenly divisible by 5. (Use decode instead of IF statement where required). Display the output on the screen using dbms_output.put_line. Data has to be input by the user.
6. Your block should read in two real numbers and tell whether the product of the two numbers is equal to or greater than 100. Display the output on the screen using dbms_output.put_line. (Use decode instead of IF statement where required). Data has to be input by the user.
7. In a PL*SQL block, create a datatype by the name of addr_type. It should contain the following components:-
   - name varchar2 (20)
   - street varchar2 (30)
   - city varchar2 (20)
   - state varchar2 (15)
8. Your block should accept the names and addresses of 4 employees in 4 different variables of datatype addr_type. Output the names and addresses of the 4 employees on the screen in the form of Labels as shown below:-

```
**************************************************************
* Name:-    Jack              **    Name:-    Scott         *
* Street:-  M.G. Road         **    Street:-  Bhosale Marg  *
* City:- Mumbai          **         City:-       Chennai    *
* State:-   Maharashtra       **    State:-   Tamil Nadu    *
**************************************************************
**************************************************************
* Name:-    King              **    Name:-    Adams         *
* Street:-  Lane No:-2        **    Street:-  P. M. Road    *
* City:- Nagpur               **    City:-    Bangalore     *
* State:-   Maharashtra       **    State:-   Karnataka     *
**************************************************************
```

9. Input a number and determine whether it is within a given range (for example, between 1 and 10). The low and high values of the range may be input by the

user rather than be fixed by the program. Display the output on the screen using dbms_output.put_line.

10. Input three positive integers representing the sides of a triangle, and determine whether they form a valid triangle. Hint: In a triangle, the sum of any two sides must always be greater than the third side. Display the output on the screen using dbms_output.put_line.
11. Check if a given a year is a leap year. The condition is:-
    year should be (divisible by 4 and not divisible by 100) or (divisible by 4 and divisible by 400.) Display the output on the screen using dbms_output.put_line. The year should be input by the user.
12. Write a program that examines all the numbers from 1 to 999, displaying all those for which the sum of the cubes of the digits equal the number itself. Display the output on the screen using dbms_output.put_line.
13. Write a PL*SQL block that reads in a minimum and maximum value for a radius, along with an increment factor, and generates a series of radii by repeatedly adding the increment to the minimum until the maximum is reached. For each value of the radius, compute and display the circumference, area, and volume of the sphere. (Be sure to include both the maximum and the minimum values.). Validate each of the input values to be sure they are positive. If the minimum is typed in place of the maximum, swap the values within the program, and continue execution. Display the results on the screen using dbms_output.put_line.
14. A table consists of the following fields:-

Invoice Number	Varchar2	4
Invoice Date	Date	
Customer Code	Number	1
Product Code	Number	1
Quantity Sold	Number	3

    There are ten customers with codes 0 to 9 and five products with codes 0 to 4. The rates of products are Rs. 15, 35, 42, 51 and 60 respectively. Write a program to find the total purchase in Rs. of each customer and total sale of each product using this table and insert these values in two other tables.
15. Write a PL*SQL block to accept a character string from the user. The user should enter a number spelt out. With the help of PL*SQL arrays, write a program for Word to number conversion up to 99 crores. The program should cater to Rs. and paise also.
    For example, if the user enters:-
    Rs. Twelve crores, Thirty Four lakhs, Fifty One thousand, Two hundred and Fifty and Seventy five paise only
    The output of your program should be: - 123451250.75
    If the user enters:-
    Rs. Nine thousand, Seven hundred and Twenty Eight only
    The output of your program should be: - 9728
16. Create the following 3 tables and insert sample data as shown:-

Ord_mst

Ord_no	Cust_cd	Status
1	C1	P

Ord_dtl

Ord_no	Prod_cd	Qty
1	P1	100
1	P2	200

Prod_mst

Prod_cd	Prod_name	Qty_in_stock	Booked_qty
P1	Floppies	10000	1000
P2	Printers	5000	600
P3	Modems	3000	200

- Write a Before Insert trigger on Ord_dtl. Anytime a row is inserted in Ord_dtl, the Booked_qty in Prod_mst should be increased accordingly.
- Write a Before Delete trigger on Ord_dtl. Anytime a row is deleted from Ord_dtl, the Booked_qty in Prod_mst should be decreased accordingly.

17. Write a stored procedure by the name of Comp_intr to calculate the amount of interest on a bank account that compounds interest yearly. The formula is:-

    $I = p (1+ r)^{y} – p$

    where:-

    I is the total interest earned.
    p is the principal.
    r is the rate of interest as a decimal less than 1, and
    y is the number of years the money is earning interest.

    Your stored procedure should accept the values of *p*, *r* and *y* as parameters and insert the Interest and Total amount into tempp table.
18. Create a stored function by the name of Age_calc. Your stored function should accept the date of birth of a person as a parameter. The stored function should calculate the age of the person in years, months and days e.g. 35 years, 3 months, 17 days. The stored function should return the age in years directly (with the help of Return statement). The months and days are to be returned indirectly in the form of OUT parameters. Write a PL*SQL block to accept the date of birth of an employee from the user, call the stored function, and display the age of the employee on the screen. Display the above results on the screen using dbms_output.put_line.
19. Write a SELECT statement to display the experience of all the employees (Sysdate – Hiredate). Your output should be as follows:-

    5 years 7 months 11 days
    9 years 3 months 16 days etc

    Don't assume that there are 365 days in a year or that there are 30 days in a month. Your solution should even take care of leap years.
20. Create a view of the Salespeople table called Commissions. This view will include only the snum and comm fields. Through this view, someone could enter or change commissions, but only to values between .10 and .20.
21. Some SQL implementations have a built-in constant representing the current date, sometimes called "CURDATE" or "SYSDATE". The word CURDATE can therefore be used in a SQL statement, and be replaced by the current date when the value is accessed by commands such as Select or Insert. We will use a view of the Orders table called Entryorders to insert rows into the Orders table. Create the Orders table, so that CURDATE is automatically inserted for odate if no value is given. Then create the Entryorders view so that no values can be given.
22. Write a command that will enable a user to pull orders grouped by date out of the Orders table quickly.
23. Write a command that puts the following values, in their given order, into the salespeople table: city – San Jose, name – Blanco, comm – NULL, cnum – 1100.
24. Write a command that removes all orders from customer Clemens from the Orders table.

Mobile and Wireless Technologies

Theory 20 hours only

Session 1:
- Basic Wireless Technologies:
- Overview of Cellular Communications:
- Basic components of a Cellular system i.e. BTS, BSC, MSC
- Protocol Stack and information flow; Call processing, MOC and MTC
- SMS Concept, Network architecture, message processing
- A Case Study: GSM

Session 2:
- Overview of CDMA:
- Concept, Network Components, Network Architecture, Digital techniques using
- Spread Spectrum
- Advantages and Disadvantages
- A Case Study

Session 3:
- Overview of 2.5 G and 3.0 G Systems:
- Need of 2.5G and 3G systems, Concepts and network architecture, Applications
- (GPRS, UMTS, 3GPP, WCDMA, HSCSD)
- Introduction to GSM and its standards,
- Services offered

Session 4:
- System architecture of GSM, Functionalities of components,
- Mobile registration process,
- Handover -- different scenarios

Session 5:
- GSM channels, traffic & control channels, Radio frequency power levels,
- Timing in advance and power control, burst structure – normal, random access,
- Frequency correction and synchronization,
- Frame structures – traffic channels, signaling frame
- Different call scenarios along with traffic and control channels,

Session 6:
- Mobile originated voice call,
- Mobile terminated voice call
- Location update procedure,
- Sending and receiving SMS
- Speech and channel coding

Session 7:
- Control channels / Logic channels
- BCCH, CCCH, FCCH, RACH, SDCCH, AGCH, PCH, SAACH, FACCH
- 2.5G / 3G systems:
- Evolution of GPRS, Core network elements, applications, architecture, QoS

Session 8 & 9:
- Mobility management, Comparison with HSCSD, Spread spectrum basics for 3G,
- Network elements, Data speeds, applications

- WLAN – various standards, a/b/g, security, applications, IAPP, Mobile IP
- Introduction, QoS aspects, integration with GPRS, Blue tooth, and 3G
- VOIP – Functionality, Shortcomings, Legal issues

Session 10:

- NFC (Near Field Communication)

iOS Programming

Theory 40 hours + Lab 44 hours

Session 1:

- Introduction to Mobile Platforms
- Overview of iPhone
- Introducing iOS core specifications
- iPhone OS Technology Layers

Session 2 & 3:

- Understanding iOS input and output
- Understanding the Basics of creating an iPhone application
- Tools for IPhone OS Development
- Xcode IDE
- Interface Builder

Session 4

- Text View Controller
- Table View Controller
- NavBar View Controller

Session 5:

- TabBar View Controller
- ToolBar View Controller
- Web View Controller

Session 6 & 7:

- Introducing the webkit
- Designing web pages for the iOS
- Using Canvas for web apps
- Building web apps with Dashcode
- Debugging iOS web pages
- SDK programming for web developers
- Using iUI for web apps
- Programming with the SDK Toolkt

Session 8 & 9:

- Handling Touch Events
- Capturing iPhone events
- Event Handling
- Creating Views Programmatically
- Introduction to OpenGL
- Creating Animations
- CSS transforms transitions and animations

Session 10 & 11:

- Device Support: Accelerometer

- Accessory (Camera and other devices)
- Device Support: Bluetooth

Session 12 & 13:

- Device Support: GPS
- Getting the User's Current Location
- Device Support: Maps

Session 14 & 15:

- Core Data, SQLite
- Address Book
- Application Preferences
- Mail Message
- Apple URL Scheme Reference
- Inter-process communication

Session 16 & 17:

- How does multithreading on iPhone OS work?
- How do I use it?
- Common multithreading on iPhone

Session 18 & 19:

- Playing Audio and Video on iPhone
- Graphics
- Signing Code for iPhone Development

Session 20:

Case Studies

Lab Assignment:

1. Hello world program on iPhone
2. Add an icon to your iPhone Xcode. Observe the maximum size of the image which you can upload.
3. Declare and define an outlet for a UITextField view using code
4. Declare and define an action using code.Using Xcode, create a new View-based Application project and name it UsingViews. Also use segmented Control to modify your project to UsingViews2.
5. Create a UIButton from code and write its Touch Up Inside event to an event handler.
6. Using Xcode, create a new View-based Application project and name it keyboardInputs.
7. Create a Tab Bar application with two Tab Bar items. When the user taps on the second Tab Bar item, display a list of movies.
8. Create a code to obtain a C-Style string from an NSString object.
9. Write the code segment to retrieve a set of rows from a table.
10. Assignment on updating data.
11. Create a service that when called stores missed calls no in database.
12. Implement a simple calculator.
13. Create an application which implements data transfer between activities.
14. Create an application which shows a list of name and images together.
15. Audio Capture Setup and Start, do and show the steps you would stop audio capture.
16. Create an application which plays an audio & video file.
17. Create an application which shows images in grid view. When you long press on image it will show you two options remove & zoom.

- When you touch on zoom it will zoom the image on another activity.
- When you touch on remove it will remove from grid view.

18. Creates a new animation whose parameters come from the specified context and attributes set
19. Create an application which show you list of name and age together when you click on the add option menu the another dialog box open fill the details and when you press ok it will load in the list view.
20. Create a service that executes when there is change in location & store the location in database with following fields-: id, latitude, longitude & time.
21. Create a service that executes every time when system boot completes and send user GPS location to a specific mobile no.
22. Create an application which shows you a grid of images. When you click on image it will rotate for 1 second.

Management Development Program

Theory 30 hours + Practice 30 hours

Session 1:

- Introduction to communication,
- Barriers to communication, Kind of communication,
- Confidence building Non-verbal Communication

Session 2:

- Fluency and vocabulary
- Synonyms
- Antonyms
- Grammar, Noun Pronoun,
- Verb, Adjective, Preposition, Conjunction

Session 3:

- Words of Idioms & phrases
- Sentence Construction
  Pronunciation,

Session 4:

Greeting,
Conversation practice,
Polite Conversation,

Session 5:

Resume Writing,
Covering letter,
Email,

Session 6:

Presentation Skill,
What is group discussion?
Interview skills, Mock interview

Session 7:

- Analogy, Series Completion (Number, Alphabet, Letter Series)
- Coding-Decoding for Number
- Alphabet and Letter
- Blood Relations

Session 8:

- Puzzle Test: Classification Type questions
- Compression Type questions
- Sequential order questions
- Section based on given conditions
- Questions involving family members

Session 9:

- Alphabet test
- Order of words
- Letter words problems
- Rule detection
- Alphabetical quibble
- Word formation
- Number
- Ranking
- Time Sequence Test
- Mathematical operations
- Logical sequence of words

Session 10:

- Arithmetic reasoning
- Logical reasoning
- Statement-Arguments
- Statement-Assumptions
- Statement-courses of Action
- Statement-Conclusions
- Deriving conclusion from passages

Session 11:

- General Aptitude
- Addition
- Multiplication
- Divisibility
- Squaring
- Cube
- HCF and LCM
- Fraction

Session 12:

- Number system
- Permutation & combination
- Probability
- Ratio & Preparation

Session 13:

- Partnership
- Percentage
- Average
- Problem on Ages
- Profit and loss

Session 14:

- Simple Interest
- Compound Interest

- Time and work
- Work and Wages

Session 15:

- Trains
- Streams Pronoun
- Alligation
- Clock
- Pipes and cisterns

Lab Practice:

1. Faculty needs to conduct GD, presentation for speaking, conducting mock interviews etc.
2. Faculty needs to conduct tests, Surprise tests, assignments etc.

7. List of Text/Reference Books

Name of Module	Title of the Book	Author/Publication	Edition	ISBN
Fundamentals of Computer and OS Concepts	Fundamentals of Computers	V. Rajaraman / PHI	5th Edition	9788120340114
	Computer Fundamentals (With CD)	Pradeep Sinha, Priti Sinha / BPB	6th Edition	9788176567527
	Operating Systems Principles	Silberschatz, Galvin / Wiley	8th	9788126520510
	Unix Concepts and Applications	Sumitabha Das / Tata Mcgraw Hill	4th	9780070635463
Objective C and OOPs concepts	Programming in Objective-C	Kochan /Pearson	4th	9788131791400
	Beginning iPhone SDK: Programming with Objective-C	Wei-Meng Lee / Wiley India Pvt. Ltd	1st	9788126525485
	Object-oriented Programming Using C++	Dehuri Satchidananda, Jagadev Alok Kumar, Rath Amiya Kumar/PHI Learning	1st Edition	9788120330856
	Object - Oriented Programming Using C++	Gopalan N. P., Sivaselvan B., Mala C / PHI Learning Private Limited	1st Edition	9788120339231
Software Development Life Cycle	Fundamentals of Software Engineering	Rajib Mall / PHI Learning	3rd Edition	9788120338197
	Software Engineering: A Practitioner's Approach	Roger S. Pressman / Tata McGraw – Hill Publication	7th Edition	9780071267823
	Software Engineering	Ian Sommerville / Pearson Publication	9th Edition	9788131762165
	Software Engineering: A Precise Approach	Pankaj Jalote / Wiley Publication	2010 Printing	9788126523115
Database Technology	Oracle Database 11g The Complete Reference	Kevin Loney / Tata McGraw - Hill	1st Edition	9780070140790
	Mastering Database Technologies	Ivan Bayross / BPB	2005 Edition	9788183331302
	Database Management Systems	Raghu Ramakrishnan, Johannes Gehrke / Tata McGraw – Hill	3rd Edition	9780071231510
Mobile and Wireless Technologies	Mobile and Wireless Design Essentials	Martyn Mallick / Wiley India Ltd.	2003 Printing	9788126503544
	Wireless and Mobile Network Architectures	Imrich Chlamtac, Yi Lin Bing/Wiley India	3rd	9788126515608
	Wireless and Mobile Networks: Concepts	Mahabaleshwar S, Kakkasageri Dr	2010 Printing	9788126520695

	and Protocols	Sunilkumar S/ Wiley		
iPhone Programming	iOS 5 Developer's Cookbook, The : The: Core Concepts and Essential Recipes for iOS Programmers	Sadun / Pearson	3rd	9788131791479
	Beginning iOS 5 Application Development	Wei-Meng Lee / Wiley India Pvt. Ltd	2012 Printing	9788126535057
	Head First iPhone Development: A Learner's Guide to Creating Objective-C Applications for the iPhone	Dan Pilone, Tracey Pilone / Shroff O'Reilly	1st	9788184048476
Management Development Program	High School English Grammar & Composition Revised Edition	Wren, Martin / S. Chand Publisher	2011 Edition	9788121900096
	Communication Skills Publication Year 2011	Sanjay Kumar, Pushp Lata / Oxford University Press	2011 Edition	9780198069324
	Professional Communication Skills	Praveen S R Bhatia / S.Chand Publishing	2011 Edition	9788121920926
	Quantitative Aptitude For Competitive Examinations	R. S. Aggarwal / S. Chand Publishing	17th Edition	9788121924986
	A Modern Approach To Verbal & Non-Verbal Reasoning	R. S. Aggarwal / S.Chand Publishing	Year 2012 Edition	9788121905510
	How to Prepare for GD and Interview (With CD)	Hari Mohan Prasad, Rajnish Mohan/TMH	3rd Edition	9780070706347

8. Evaluation Guidelines

8.1 Evaluation

Evaluation is a necessary and essential part of conducting the C-DAC Certified iPhone Programming, as it provides important feedback and inputs to both the institute as well as the student. The institute gets an idea about the relative performance of each student, which also serves as feedback about the design and conduct of the programme. The student gets a clear picture of his academic standing, individually and in comparison to his fellow students.

In order to ensure timely and efficient evaluation and certification of all students, the following guidelines are being issued and should be followed religiously.

8.2 Evaluation Methodology

8.2.1 Each centre should have a Designated Responsible Member (DRM) for Evaluation.

8.2.2 The DRM Evaluation would be responsible for coordinating all activities relating to evaluation at the training centre and for communicating with CDAC ACTS, Pune.

8.2.3 Evaluation is a compulsory part of the process of obtaining C-DAC Certified iPhone Programming certificate. All students are required to pass in each subject of the course in order to be eligible to receive the C-DAC Certificate.

8.2.4 The faculty of every subject should outline the objectives of the evaluation to be conducted for that particular subject, so as to enable the student to prepare himself/ herself properly.

8.2.5 The performance of students is constantly evaluated through surprise quizzes, hourly examinations, assignments throughout the term, submission of term reports, presentations and final examinations at the end of the course.

8.2.6 Mode of exams will be in online / offline, but prior information will be given by C-DAC, ACTS about the mode of the exam and it will be final.

8.3 EVALUATION METHODS

8.3.1 Course End Evaluation

After completion of the all subjects, a written examination CEE (Course End Examination) will be held, which will test the knowledge of the students of each subject and it is a compulsory part of the evaluation. Conducting CEE involves performing duty with responsibility. A small mistake in the process may hamper the whole system. Everyone has to play their role in an effective manner. It is a joint effort work which has to be carried out in a combined way. Right from receiving question paper from ACTS, C-DAC to sending the OMR answer sheet (in case of offline exam) and the response file (in case of online exam) for evaluation dealt with lot of responsibility.

ACTS, C-DAC in its pursuit of excellence, believes in providing a congenial atmosphere to the students during all exams in order to get them to perform at their optimum level. However, there are certain norms which the students are expected to be aware of and observe both in letter and spirit. These norms are:

8.3.1.A Impersonation may lead to permanent expulsion from the Institute.

8.3.1.B Cell phones are strictly prohibited in the exam hall/room.

8.3.1.C Valid ID card is mandatory for entry to the exam room / hall.

8.3.1.D Punctuality is most important at all times. Students are expected to check their exam location and be seated at least 10 minutes prior to the exam time.

8.3.1.E In case of offline exam, as per ACTS, C-DAC policy all question papers are to be returned along with the answer script.

8.3.1.F Students are required to bring their own stationary as no lending or borrowing is permitted during examination.

8.3.1.G Programmable calculators or any other kind of electronic devices are strictly prohibited inside the exam area.

8.3.1.H Indiscipline in the exam hall/ room will not be tolerated.

8.3.1.I Possession of any written material related to the subject or communication with their fellow students, will result in disciplinary actions.

8.3.1.J A student must score a minimum of 40 percent marks, in order to successfully clear the course.

8.3.1.K It is recommended that the students should ensure 100% attendance for each course. 10% absences are permissible, only in case of illness, or emergencies. These have to be approved by the Centre Head. Approval is contingent upon the evidence provided.

8.3.1.L There will be 150 questions to answer in 3 hours duration in CEE as per the following distribution mentioned in Table – 1.

Table – 1

Sl. No.	Module Name	Hours	No. of Questions
1	Fundamentals of Computer & OS Concepts	20	10
2	Objective C and OOPs concepts	60	35
3	Software Development Life Cycle	16	10
4	Database Technology	20	10
5	Mobile and Wireless Technologies	20	10
6	iOS Programming	84	45
7	Management Development Program	60	30
8	Project	40	Grade
	Total	**320**	**150**

8.3.2 GENERAL GUIDELINES FOR AWARD OF GRADES:

The marks of obtained in the CEE shall be calculated to get total marks out of 100. The rounding off shall be done on the higher side. The grades shall be awarded on the basis of cut off in the absolute marks, as mentioned in Table – 2.

Table 2

Lower range of marks	Grade	Upper range of marks
91	≤ A+ <	100
81	≤ A <	90
71	≤ B+ <	80
61	≤ B <	70
51	≤ C+ <	60
41	≤ C <	50
0	≤ F <	40

8.3.3 Guidelines of CEE:

CEE will be conducted normally before the commencement of Project work of the course. The written examination should be of 180 minutes duration. It should consist of objective questions. A typical objective type exam paper should contain the following types of questions: –

- Multiple choice

- Yes or No
- True or False

Objective questions are useful in testing the recognition and recall abilities of students. They also help in keeping the exam short and easier to evaluate.

For the pure objective type question papers, there will be 150 objective type questions with 4 maximum answer options having only one correct option. The value of each objective type question is of one mark only. There will not be any negative marks for the wrong answers given by the students.

8.3.4 Guidelines for setting Question Papers:

While setting the question papers for theory Exam the following weightages should be assigned as per the difficulty level of the questions.

Levels	Requirements	Weightage
Level A – Easy	Requires elementary knowledge which may be obtained by attending all lectures and completion of mandatory lab assignments	25%
Level B – Intermediate	Requires thorough study of all course material, attendance at all lectures and completion of mandatory assignments	50%
Level C – Difficult	Requires study and lab work beyond the prescribed course material and mandatory assignments	25%

8.4 Guidelines for generating questions:

8.4.A Question paper setter has to use sample paper format provided by C-DAC, ACTS Pune

8.4.B Mention the subject name without fail.

8.4.C Language of the question should be easy to understand.

8.4.D The answers must have relevant objective type choices and “only one” correct answer.

8.4.E The questions must be prepared by referring appropriate books, reference books, reference material, and course material having good information.

8.4.F The question must be created by the domain expert afresh and should not be copied directly from any book, website, existing previous question papers etc.

8.4.G The question should be unique and should have not been published anywhere.

8.4.H Please mention the source of the question wherever possible, as it may help us in referring the same for detailing if required.

8.4.I The caliber of the question should suffice the growing need of competition.

8.4.J The question paper should have questions covering the entire syllabus.

8.4.K The questions have to be typed in MS Word with “Arial” having letter size 12 point. Do not bold any letter, word or sentence in any part of the question paper.

8.4.L It is essential to give password to the word document and send/tell the password separately.

8.4.M It is essential that utmost care is taken at your end to maintain the secrecy of the soft copy at all time.

8.4.N An expert team will review all questions. The questions will be filtered as per following:

- If the question is incomplete
- If the answer of the question is wrong
- If the question is not there in the syllabus

- If the question appears more than once
- If the question is too lengthy
- If the question is irrelevant
- If the options to the questions are irrelevant

8.4.1 Template for generation of Questions

Date:

Question generated by: Mr. /Ms.

Subject Name:

Q. No.
Question: <Text of the question>

Answer Choices
A:
B:
C:
D:

Difficulty Level: Easy / Intermediate / Difficult

Reference: (Name of books)
(If question taken from book) (Mention name of the book, author, ISBN)
Total Number of Questions Generated: _____

8.4.2 Template for Answer Key:

Module name:			
Question No.	**Answer Keys**	**Question No.**	**Answer Keys**
1		76	
2		77	
3		78	
4		79	
5		80	
6		81	
7		82	
8		83	
9		84	
10		85	
11		86	
12		87	
13		88	
14		89	
15		90	
16		91	
17		92	
18		93	

19		94	
20		95	
21		96	
22		97	
23		98	
24		99	
25		100	
26		101	
27		102	
28		103	
29		104	
30		105	
31		106	
32		107	
33		108	
34		109	
35		110	
36		111	
37		112	
38		113	
39		114	
40		115	
41		116	
42		117	
43		118	
44		119	
45		120	
46		121	
47		122	
48		123	
49		124	
50		125	
51		126	
52		127	
53		128	
54		129	
55		130	
56		131	
57		132	
58		133	
59		134	
60		135	
61		136	
62		137	
63		138	
64		139	
65		140	
66		141	
67		142	

68		143	
69		144	
70		145	
71		146	
72		147	
73		148	
74		149	
75		150	

8.4.3 Evaluation of answer papers:

For Offline mode: Use of OMR sheets will be useful for processing the result of multiple choice exams. OMR is an effective way to collect data, process for the result and also it takes less time with greater accuracy in less effort. Centres need to follow the best way for scanning the OMR sheets, process the result and publish the result. Centres which are not using OMR can user OCR to conduct the exams and evaluate the students. Centre which are not using OMR or OCR can evaluate the students manually and process the result.

For Online mode: Course end exam will be through online s/w. Evaluation will be through that Exam s/w.

If a student requests for re-evaluation then the student has to pay `150 /- and it should be routed through training centre. The Re-evaluation fee should be paid to respective C-DAC training Centres, in case of Authorized Training Centres associated to C-DAC, Pune, payment to be made in favour of "C-DAC, ACTS" and payable at Pune. (This is applicable only for theory exam)

8.5 Moderation:

Grace marks would be awarded as per the methodology below:

8.5.1. Maximum of 4% of total term end theory exam marks can be awarded to a candidate.

Sr. No.	Name of the Course	Total Marks	Maximum grace marks
1	C-DAC Certified iPhone Programming	150	6

On completion of the moderation exercise the revised marks should be updated in the marks database.

8.6 Re-examinations:

The following conditions will be applicable for the course end re-exam:

8.6.1. Students who do not appear for an exam on the scheduled date will not have an automatic right to re-examination. Only those students who, in the opinion of the centre/course coordinator have a genuine reason for being absent may be allowed to appear for a re-exam.

8.6.2. Students who have failed an exam may be allowed to appear for a re-exam.

8.6.3. The re-exam should be conducted following the same process as the regular examination.

8.6.4. Students, who failed/remained absent in the Course End Examination conducted by C-DAC, shall be allowed to appear in the re-examination only once.

8.6.5. Students who remain absent or fail in the re-examination will not get any further chance for appearing for a third attempt or further. In such case the candidate can receive the Performance Statement and the certificate of participation without any grade.

8.6.6. On evaluation of their answer sheets 20% of the marks obtained by the students will be deducted (towards de-rating for re-examination) for arriving at the final score, i.e. in order to clear the module test the student has to score a minimum of 50% marks instead of 40%.

8.7 Project Module:

8.7.1. Project work should be start at the time of Java Programming module and database design concept should be complete by that time.

8.7.2. After that students should be ready with all mandatory documents with database design and then completion of all teaching modules they can do the project.

8.7.3. Performance in the Project module will be awarded in grade. The Project grade will be mentioned separately on the certificate & will have no effect on the overall grade obtained by a student.

8.7.4. Students may do industry-sponsored projects, but will be required to do the project work within the centre.

8.7.5. Evaluation of the Project module will take place as following:

8.7.5.1. Internal evaluation will be take place at mid of the module

8.7.5.2. External evaluation will take place at the end of the module

Based on both evaluations, final grade will be awarded & communicated to C-DAC ACTS, Pune

8.7.6. **Guidelines for Project Evaluation**

Evaluation of Project work needs to be carried out as per the following guidelines:

a. Literature study.
b. Submission of abstract for their colloquium/seminar/project work along with the references.
c. Submission of the detailed work report
d. Two presentations each for 15 minutes on the work done restricted to 15 – 20 slides followed by evaluation.
e. The evaluation for 100 marks will be splited up as follows:

i.	Literature survey	– 10
ii.	Contents of the project work	– 20
iii.	Contents Flow of Presentation	– 15
iv.	Communication and Presentation Skills	– 20
v.	Depth of Knowledge in the topic	– 15
vi.	Viva Voce	– 15
vii.	Attendance	– 5

f. Soft copy of the presentation should be submitted to C-DAC.

8.8 Ensuring Security of Evaluation data/records:

8.8.1. Ensure that all data relating to evaluation of students is stored in a secure place that cannot be accessed by unauthorized personnel.

8.8.2. All question papers must be prepared and stored in a separate area specifically designated for the purpose.

8.8.3. Whenever any external faculty sets a question paper, ensures that he should follows the guidelines given by C-DAC ACTS Pune.

8.8.4. Ensure that only one copy of any question paper is prepared in physical (printed) form for review and revision.

8.8.5. When the question paper is finalized, print out one master copy and gets it signed by the paper setter, Reviewer and DRM Evaluation.

8.8.6. Prepare required number of photocopies of the question paper and store them in a safe and secure location before the exam.

8.8.7. The data relating to evaluation of students, such as soft copies of question papers and answer keys, student marks database and performance statements etc. must be kept in a separate domain/directory which is accessible only to authorized personnel. Ensure that the data is regularly backed up.

8.8.8. The question papers for the theory as well as the laboratory examinations at all the centres will be set by CDAC, ACTS Pune. The centres according to guidelines provided by C-DAC, ACTS Pune, will conduct the evaluation of the laboratory and assignments locally.

Note: The Evaluation Guidelines, Rules and Regulations issued by C-DAC, ACTS – Pune from time to time shall be binding on all the centers and all the students. C-DAC, ACTS, Pune reserves the right to add, modifies or deletes any or entire contents of this document at any point of time without giving any notice. It's the responsibility of the centre coordinator to inform such changes to the students in form of a formal notice with a duly signed copy to C-DAC, ACTS, Pune.

9. Requirements (S/W and H/W)

Computing Facilities for Certificate Course in iPhone Programming		**Yes/No**
A. Servers		
1. Unix / Linux Server		
2. Windows Server 2008		
3. Servers configured for various modules		
Severs Configuration		
1. Processor (min 3.2 Ghz)		
2. RAM (min 4 GB)		
3 HDD (min 500 GB)		
4. Network Card		
5. 4 USB , optical mouse, us Keyboard.		
6. DVD Drive		
B. Clients Machines Configuration		
1. Processor (Min 3.2 Ghz)		
2. RAM (Min 2 GB)		
3. HDD , SATA/ IDE (min 250 GB)		
4. PCI Network Card 1Gbps, CAT 5 Ethernet		
5. Multimedia Kit		
C. Network		
1. Gigabit Switch		
2. CAT-5 Cabling with RJ-45 connectors		
3. Patch Cables		
D. Communication and Internet		
1 Internet Access		
2. DSL /fiber line		
3. Minimum required bandwidth 1Mbps		
E. Printers		
1. Laser Printer		
F. Additional Lab Equipments		
1. Amplifier Speakers, Headphones & Mikes		
2. Hi-Lumen OHP		
3. Video Projector (SVGA/HDMI/ Svideo Compatible)		
4. TWAIN Compliant Color Scanner		
G. Module Specific Software Environments, Operating Systems and Hardware		
1. iOS Programming	iPhone 5S	
2. Mobile Wireless Programming and Application	1. Blue tooth USB Adapter 2. GPRS enabled SIM card 3. Wireless Access Point	
3. Fundamentals of Computer & OS Concepts	gcc in Open Suse 13.1	
4. Objective C and OOPs Concepts	gcc in Open Suse 13.1	
5. Database Technology	Oracle 11g Client	
H. Operating System Software Common For all Course modules		
1. For Windows platform–**Windows 7 or 8**, 2. For Linux platform–**SUSE Linux** (latest), 3. **MAC OS Lion 10.7** with xcode 4.2 or Latest Version		**Yes/No**