Six Months Course in Computer Programming
(NIELIT CERTIFIED)

☐ Programming thru C

☐ Introduction to Object oriented Programming thru C++ or JAVA

☐ Visual Basics using .Net

Introduction to Programming
☐ The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation

Algorithms for Problem Solving .
☐ Exchanging values of two variables, summation of a set of numbers, Decimal Base to Binary

☐ Base conversion, Reversing digits of an integer, GCD (Greatest Common Division) of two numbers, Test whether a number is prime, Organize numbers in

☐ ascending order, Find square root of a number, factorial computation, Fibonacci

☐ sequence, Evaluate 'sin x'as sum of a series, Reverse order of elements of an

☐ array, Find largest number in an array, Print elements of upper triangular matrix, multiplication of two matrices, Evaluate a Polynomial

Introduction to 'C' Language
☐ Character set, Variables and Identifiers, Built-in Data Types, Variable Definition, Arithmetic operators and Expressions, Constants and Literals, Simple assignment statement, Basic input/output statement, Simple 'C' programs.

Conditional Statements and Loops
☐ Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, structured Programming .

Arrays
☐ One dimensional arrays: Array manipulation; Searching, Insertion, Deletion of an

☐ element from an array; Finding the largest/smallest element in an array; Two dimensional arrays, Addition/Multiplication of two matrices, Transpose of a square matrix; Null terminated strings as array of characters, Standard library string functions

Functions
☐ Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, Passing arguments to a Function: c

☐ all by reference, call by value, Recursive Functions, arrays as function arguments. Storage Classes Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in a multiple source files: extern and static

Programming Thru C++
☐ Basics of C++, Elementary Data Structures

☐ Basics of C++:

Structure of a programVariables. Data Types.ConstantsOperators, Basic Input/Output,
Control Structure , Functions, Compound Data Types: Arrays, Pointers,DynamicMemory,ObjectOrientedProgramming:Classes,Encapsulation,Abstracon, inheritance,Polymorphism, Representation of arrays: single and multidimensional arrays. Address calculation using column and row major ordering. Various operations on Arrays, Vectors. Application of arrays: Matrix multiplication, Sparse polynomial representation and addition, Stacks and Queues : Representation of stacks and queues using arrays and linked-list. Circular queues, Priority Queue and D-Queue. Applications of stacks: Conversion from infix to postfix and prefixexpressions, Evaluation of postfixexpression using stacks. Pointers: Definition, Pointer Arithmetic,Array of pointers, Arrays in terms of pointers. Linked list: Singly linked list; operations on list, Linked stacks and queues.Polynomial representation and manipulation using linked lists.Circular linked lists, Doubly linked lists. Generalized list

structure.Sparse Matrix representation using generalized list structure, stacks, queues.
 Abstract Data types Stacks and Queues

Definition of ADT, Stack ADT (array implementation), FIFO queue ADT (array implementation)
 Trees

Binary tree traversal methods: Preorder, In-order,Post-ordered traversal. Recursive Algorithms for above mentioned Traversal methods. Representation of trees and its applications : Binary tree representation of a general tree. Conversion of forest into tree. Threaded binary trees. Binary search tree. : Height balanced (AVL) tree, B-trees
 Searching, Sorting and Complexity

Selection sort, Insertion sort, Bubble sort, Quick sort, merge sort , Heap sort, Radix sort and their complexity, Searching: Sequential search, Binary Search, Binary Search Tree, ASVL trees, B trees, Searching , sorting and complexity,
Searching : Sequential and binary searches, Indexed search, Hashing Schemes. Sorting : Insertion, selection, bubble, Quick, merge, radix, Shell, Heap sort, comparison of time complexity.
 Graphs

Graph representation : Adjacency matrix, Adjacency lists, Traversal schemes : Depth first search, Breadth first search. Spanning tree : Definition, Minimal spanning tree algorithms. Shortest Path algorithms

Structures and Unions
☐ Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions
☐

Pointers
☐
☐ Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation.
☐ Self Referential Structures and Linked Lists Creation of a singly connected linked list, Traversing a linked list, Insertion into a linked list, Deletion from a linked list

Introduction to Object Oriented Programming
Thinking Object-Oriented
Why Is OOP Popular? A New Paradigm, A Way of Viewing the World. Why Is OOP Popular? A New Paradigm, A Way of Viewing the World.
Abstraction
☐ Layers of Abstraction, Other Forms of Abstraction

Classes and Methods
Encapsulation, Class Definitions, Methods.
Messages, Instances, and Initialization
☐ Message-Passing Syntax, Statically and Dynamically Typed Languages,
☐ Accessing the Receiver from Within a Method, Object Creation, Pointers
☐ and Memory Allocation, Constructors{Constant Values}, Destructors and Finalizers.

Inheritance and Substitution
☐ An Intuitive Description of Inheritance, Inheritance in Various Languages, [Subclass, Subtype, and Substitution], Overriding and Virtual Methods, Interfaces and Abstract Classes, Forms of Inheritance, The Benefits of Inheritance, The Costs of Inheritance. Examples (Language independent)

Static and Dynamic Behavior
☐ Static versus Dynamic Typing, Static and Dynamic Classes, Static versus Dynamic Method Binding.

Multiple Inheritance
☐ Inheritance as Categorization, Problems Arising from Multiple Inheritance, Inner Classes.

Polymorphism and Software Reuse
☐ Polymorphism in Programming Languages, Mechanisms for Software Reuse, Efficiency and Polymorphism, Will Widespread Software Reuse Become Reality?

Overloading and Overriding
☐ • Type Signatures and Scopes, Overloading Based on

 Scopes, Overloading Based on Type Signatures, Redefinition, Notating Overriding, Replacement

 versus Refinement, Deferred Methods, Overriding versus Shadowing, Covariance and Contra variance.

Introduction to Programming Language thru JAVA
An Introduction to Java
 Java as a Programming Platform, The Java "White Paper" Buzzwords,
 Java and the Internet, A Short History of Java, Common Misconceptions About Java.

The Java Programming Environment
 Installing the Java Development Kit, Choosing a development Environment, Using the Command-Line Tools, Using anIntegrated Development Environment, Compiling and Running Programs from a Text

Editor, Running a Graphical Application, Building and Running Applets.
Fundamental Programming Structures in Java
• A Simple Java Program, Comments, Data Types, Variables, Operators, Strings, Input and Output, Control Flow, Big Numbers, Arrays.
Objects and Classes
• Introduction to Object-Oriented Programming, Using Predefined Classes, Defining Your Own Classes, Static Fields and Methods, Method Parameters, Object Construction, Packages, Documenation Comments, Class Design Hints.
Inheritance
• Classes, Superclasses, and Subclasses, Object: The Cosmic Superclass, Generic ArrayLists, Object Wrappers and Autoboxing,Reflection, Enumeration Classes, Design Hints for Inheritance.
Introduction to GUI
Interfaces, Object Cloning, Interfaces and Callbacks, Inner Classes, Proxies. AWT Architecture, Light-Weight vs Heavy-Weight, AWT Event Model, AWT Event Hierarchy & Event Handling, Using Top-Lev
els, components and containers, Introduction to Layouts, Focus Architecture.
Graphics Programming
• Java2D Rendering Model, Strokes & Fills, Geometries, Fonts and Text Layout, Transformations, Display and manipulation of Images and offscreen buffers, Using Color, Printing through Java, Doing More with

Images using Image IO, Hardware Acceleration and Active Rendering techniques.

User Interface Components with Swing

• The Model-View-Controller Design Pattern, Introduction to Layout Management, Text Input, Choice Components, Menus, Sophisticated Layout Management, Dialog Boxes.

Deploying Applets and Applications

• Applet Basics, The Applet HTML Tags and Attributes, Multimedia, The Applet Context, JAR Files, Application Packaging, Java Web Start, Storage of Application Preferences.
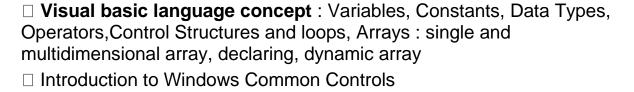
Exceptions and Debugging

• Dealing with Errors, Catching Exceptions, Tips for Using Exceptions, Logging, Using Assertions, Debugging Techniques, Using a Debugger.

Streams and Files

• The Complete Stream Zoo, ZIP File Streams, Use ofStreams, Object Streams, File Management, New I/O, Regular Expressions.

Database Programming

• The Design of JDBC, The Structured Query Language, JDBC Installation, Basic JDBC Programming Concepts, Query Execution, Scrollable and Updatable Result Sets, Metadata, RowSets, Transactions, Advanced Connection Management, Introduction to LDAP.

Microsoft .NET Framework.

☐ The .NET Framework components

☐ The Common Language Runtime (CLR), Environment, The .NET Framework class Library.

☐ Getting Started with Visual Basic .net IDE : Set up of work environment, start page, the menu system, toolbars, the new project dialog box, graphical designers, code designers, intellisense, the object explorer, the toolbox, the solution explorer, the class view window, the properties window, the dynamic help window, the server explorer, the output window, the command window.

☐ **Visual basic language concept** : Variables, Constants, Data Types, Operators,Control Structures and loops, Arrays : single and multidimensional array, declaring, dynamic array

☐ Introduction to Windows Common Controls

☐ **Working with Form** :Properties : appearance, behaviour, layout, windows style etc, methods and events Differentiate procedure oriented, object oriented and event driven programming Inputbox, Messagebox Working with Common Tool Box Controls: Label & button (Properties: flatstyle, image, imagealign etc.), Textbox (Properties: autosize, maxlength, multiline, readonly, wordwrap etc.), NumericUpDown (textalign, updownalign, value, interceptarrowkeys, decimalplaces, increment, maximum, minimum etc.) Check Box (autocheck, checked, checkaligned,checkstate, threestate etc.), Radio Button (check aligned, check, aut o check etc.), Group Box(gridsize:width, height etc.) control and all important methods and events

☐ **Working with other controls of toolbox** : Date Time Picker, List Box, Combo box, Picture Box, Rich Text Box, Progress bar, Masked Text box, Link Label, Checked List box, Scroll Bars, timer.

☐ **Working with Menus**: creating menu, inserting, deleting, assigning short cut keys,

☐ popup menu

☐ **Dialog Boxes:** OpenFileDialog, SaveFileDialog, FontDialog, ColorDialog, PrintDialog

☐ Sub Procedures and functions : declaring, passing and returning arguments, exiting from it, pass by value and pass by ref

Exception Handling : Structured Error Handling (Try ....Catch ....finally), Unstructured Error Handling (On error go to line,goto 0, goto -1, resume next)

☐ Multiple document interface (MDI) : MDI Parent form and child form

☐ Inbuilt Functions: Inbuilt functions and String manipulation functions of VB.Net

☐ ADO .NET Object Model: Data provider(connection, command, data reader, data

☐ adapter, data readers) Dataset (data table collection (data table, data rows, data columns, data constraints), data relation collection)

☐ **ADO .NET Programming** :Creating a Database Application, Creating Connection to

☐ a Database using ADO.NET , Populating Data in ADO.NET, Browsing Records, Datagrid view, Editing, Saving, Adding and Deleting

☐ Records using bounded and unbounded