GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

Course Curriculum

OPERATING SYSTEMS
(Code: 3330701)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering, Information Technology	3rd Semester

1. RATIONALE

An operating system is the core software of any computer system. This is the basic software or platform on which other software work. Every student of computer science and IT must therefore understand basic structure of an operating system. After learning this subject student will be able to discriminate between various types of operating systems, its processor, processes, and memory and file management. The subject also emphasis on Linux utilities and scripting**.**

2. COMPETENCY (Programme Outcome (PO) according to NBA Terminology):

The course content should be taught and implemented with the aim to develop different types of skills so that students are able to acquire following competency:

- **To install & configure various Operating Systems.**

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	150
3	0	2	5	70	30	20	30	

Legends: L - Lecture; **T -** Tutorial/Teacher Guided Student Activity; **P -** Practical; **C -** Credit; **ESE** - End Semester Examination; **PA** - Progressive Assessment

4. COURSE DETAILS

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit – I Operating System Concepts	1a. Explain different operating system	1.1 Need of operating system 1.2 Evolution of operating system
	1b. Explain types of operating system	1.3 Operating systems i. Batch ii. Multi programming iii. Time Sharing iv. Real Time v. Multitasking vi. Multithreading 1.4 Operating System Services 1.5 Case study i. Linux ii. Windows 7
Unit – II Processor & Process Management	2a. Describe process model 2b. Describe process state	Process and Process management i. Process model overview ii. Programmers view of process iii. Process states
	2c. Compare processor scheduling algorithm.	2.2 Process and Processor Scheduling i Scheduling Criteria ii First Come First Serve iii Round Robin iv SJF v SRTN
	2d. Compare different scheduler 2e. Describe race condition & mutual exclusion	2.3 Schedulers i Inter Process communication & a. synchronization ii Race condition iii Mutual Exclusion iv Monitors
	2f. Identify Deadlocks 2g. Apply Deadlock recovery procedure	2.4 Dead lock i Prevention ii Avoidance iii Detection and recovery
Unit – III Memory Management	3a. Describe memory management	3.1 Memory management
	3b. Differentiate Contiguous and Non-contiguous memory 3c. Differentiate physical and virtual primary memory	3.2 Contiguous allocation i Partitioned memory allocation ii Fixed & variable partitioning iii Swapping iv Relocation v Protection and Sharing 3.3 Non contiguous allocation i Page allocation ii Segmentation iii Virtual Memory

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit – IV File Management	4a. Apply file management concepts in Operating System	4.1 File management i. User view of file system ii. Attributes and operations iii. File system design iv. Disk space
	4b. Explain Directory structure of Operating System	4.2 Directory structure
	4c. Describe Disk organization	4.3 Disk Organization i. Physical structure ii. Logical structure iii. Addressing
	4d. Implement file system security.	4.4 Security and Protection mechanism
Unit – V Linux Basics	5a. Install Free & Open Source Software / Open source Operating System	5.1 Overview of Linux 5.2 Installation and upgrade
	5b. Test and Execute basic Linux commands	5.3 Introduction to shell and commands i. Commands: pwd, cd, mkdir, rmdir, ls, cat, cp, rm, mv, wc, split, cmp, comm, diff, head, tail, grep, sort, apt-get install, apt-get remove
	5c. Test and Execute shell commands in a script	5.4 Editing files with "vi", "vim", "gedit", "gcc" 5.5 Linux Shell Basic shell scripts

5. SUGGESTED SPECIFICATION TABLE WITH HOURS & MARKS (THEORY)

Unit	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Operating System Concepts	06	04	06	0	10
II	Processor & Process Management	12	06	10	04	20
III	Memory Management	10	06	08	02	16
IV	File Management	06	04	06	0	10
V	Linux Basics	08	02	04	08	14
Total		**42**	**22**	**34**	**14**	**70**

Legends: R = Remember; U = Understand; A = Apply and above levels (Bloom's revised taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

6. SUGGESTED LIST OF EXERCISES/PRACTICAL

The practical/exercises should be properly designed and implemented with an attempt to develop different types of practical skills **(Course Outcomes in psychomotor and affective domain)** so that students are able to acquire the competencies (Programme Outcomes). Following is the list of practical exercises for guidance.

Note: Here only Course Outcomes in psychomotor domain are listed as practical/exercises. However, if these practical/exercises are completed appropriately, they would also lead to development of **Programme Outcomes/Course Outcomes in affective domain** as given in a common list at the beginning of curriculum document for this programme. Faculty should refer to that common list and should ensure that students also acquire those Programme Outcomes/Course Outcomes related to affective domain.

S. No.	Unit No.	Practical/Exercise (Course Outcomes in Psychomotor Domain according to NBA Terminology)	Apprx. Hrs. Required
1	**I**	Install & test different types of Operating System & compare its features.	2
2	**II**	Compare various process scheduling algorithm	2
3	**V**	Test and run basic unix commands.	2
4		Test and run Advanced unix commands.	2
5		Test commands related with File editing with Vi, Vim, gedit, gcc.	2
6		Create a shell script to print ”Hello”.	2
7		Create a Shell script to read and display content of a file.	2
8		Create a Shell script to read from command line.	2
9		Create a Shell script to append content of one file to another	2
10		Create a Shell script to accept a string in lower case letters from a user, & convert to upper case letters.	2
11		Create a Shell script to find numbers of characters, words & lines of a given input file.	2
12		Create a Script to reverse a string and display it.	2
13		Create a Script to check a string is palindrome.	2
14		Create a Shell script to add two numbers.	2
15		Create a shell script to reverse the digits of a given 5-digit number. (for eg. , if the no. is 57429 then answer is 92475).	2
		Total	**30**

7. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:

- PowerPoint Presentation
- Seminar based Presentation
- Case study

8. SPECIAL INSTRUCTIONAL STRATEGY (If Any)

Concepts should be explained thoroughly in theory sessions and should be implemented in laboratory appropriately along with the problem solving. Concept should be developed by giving problems to students as assignments and in tutorials.

9. SUGGESTED LEARNING RESOURCES

(A) List of Books:

S. No.	Title of Books	Author	Publication
1	Operating systems	Dhamdhere	MGH
2	Unix Concepts And Application	Sumitabha Das	MGH
3	Modern Operating System 3rd Edition, 2008	Andrew Tanenbaum	PHI
4	Operating System Concepts, 3rd Edition	James Peterson Wesley Abraham Silberschatz	JOHN WILEY & SONS. INC
5	Operating Systems, 2010 Edition	Sibsankar Haldar	Pearson Education
6	Operating System, 2005 Edition	Milan Milenkovic	MGH
7	Operating Systems concept based approach (3rd Edition)	Dhananjay M.	MGH
8			
9	Linux –Application and administration, 2009 Edition	Ashok Kumar Harnal	TMH

B. List of Major Equipment/Materials

i Linux based Host machines (Free & Open Source Software or Open source)

ii Computers with latest hardware configuration

C List of Software/Learning Websites

i Operating System concepts: http://nptel.iitm.ac.in/courses/Webcourse-contents/IISc-ANG/Operating%20Systems/New_index1.html

ii Linux basics: www.freeos.com/guides/lsst

iii Linux basics: www.linuxcommand.org/writing_asell_scripts.php

iv Linux basics: www.distro.ibiblio.org/damnsmall/current/dsl-4.4.10-embedded.zip

10. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

- **Prof. Manoj P. Parmar**, In-charge Head of Department, Information Technology, Government Polytechnic, Ahmedabad.
- **Prof. Parvez K. Faruki**, Lecturer, Information Technology, Government Polytechnic, Ahmedabad.
- **Prof. (Mrs.) Harsha P. Chauhan**, In-charge Head of Department, Information Technology, Government Polytechnic for Girls, Ahmedabad.
- **Prof. Darshan M. Tank**, In-charge Head of Department, Information Technology, Lukhdhirji Engineering College (Diploma), Morbi

Coordinator and Faculty Members from NITTTR Bhopal

- **Dr. Shailendra Singh**, Professor & Head Dept. of Computer Engineering and Applications,
- **Dr. K. J. Mathai**, Associate Professor Dept. of Computer Engineering and Applications,

GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

Course Curriculum

PROGRAMMING IN C++
(Code:3330702)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering	3rd Semester

1. RATIONALE

This course intends to teach the students about basic concepts of Object-Oriented Programming (OOP) and C++. Large programs are probably the most complicated entities ever created by humans. Because of this complexity, programs are prone to error and software errors can be expensive and even life-threatening. Object-oriented programming offers a new and powerful way to cope with this complexity and act as the backbone to all other courses that are based on Object Oriented concept. Therefore by learning this course senicierly the students will be able to develop programs in 'C++' using Object Oriented Programming Concepts.

2. COMPETENCY (Programme Outcome (POs) according to NBA Terminology):

The course content should be taught and implemented with the aim to develop various types of related skills leading to the achievement of the following competency

- **Develop programs in 'C++' using Object Oriented Programming Concepts.**

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	
3	2	2	7	70	30	20	30	150

Legends: L -Lecture; **T** -Tutorial/Teacher Guided Student Activity; **P** -Practical; **C** -Credit;**ESE**-End Semester Examination; **PA** -Progressive Assessment

4. COURSE DETAILS

Unit	MajorLearning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit – I Principles of Object Oriented Programming	1a. Differentiate procedure and object oriented languages	1.1 Procedure oriented Programming 1.2 Object oriented programming paradigm 1.3 Basic concepts of Object Oriented Programming 1.4 Advantages of Object Oriented Programming 1.5 Object Oriented Languages 1.6 Applications of Object Oriented Programming
	1b. Explain the general structure of C++ Language	1.7 C++ Concepts 1.8 Structure of C++ program 1.9 Applications of C++
	1c. List different data types available in C++	1.10 Basic Data types in C++ 1.11 User defined Data types 1.12 Derived Data types
	1d. Inititalize Data using variables and develop simple C++ programs	1.13 Defining Constants 1.14 Declaration of variables and Dynamic initialization of variables 1.15 Reference variables
	1e. Differentiate various operators	1.16 Operators in C++ 1.17 Scope Resolution Operators 1.18 Member dereferencing Operators 1.19 Memory Management Operators and Manipulators 1.20 Type cast Operator
Unit– II Functions in C++ and Working with objects	2a. Develop programs using functions	2.1 The Main Function 2.2 Function prototyping 2.3 Call by Reference and Return by Reference
	2b. Develop programs using inline functions	2.4 Inline functions
	2c. Define functions using default, constant, arguments, function overloading	2.5 Default Arguments 2.6 Constant Arguments 2.7 Function Overloading
	2d. Develop Simple Programs using class and objects, array of objects, friend functions, passing and returning objects	2.8 Classes and Objects : 2.9 Overview of C structure 2.10 Defining Class and Creating Objects 2.11 Memory Allocation for Objects 2.12 Defining Member function 2.13 Making an outside function Inline 2.14 Nesting of Member functions 2.15 Private Member functions 2.16 Arrays within a Class

Unit	MajorLearning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
	2e. Differentiate static members and normal members	2.17Static Data member and Static Member functions, 2.18Array of Objects, 2.19Passing Objects as an Argument, Returning Object, 2.20Friend function, Pointer to members
Unit– III Constructor and Destructor	3a. Define constructor & destructor	3.1 Constructor Concepts 3.2 Destructor
	3b. Develop program using constructor and destructor	3.3 Parameterized Constructor, 3.4 Multiple Constructors in a Class, 3.5 Constructor with Default Arguments, 3.6 Copy Constructor, 3.7 Dynamic Constructor
Unit– IV Inheritance	4a. Define Inheritance	4.1 Concepts of Inheritance
	4b. List the applications of inheritance, types of inheritance and develop programs using single, multilevel and multiple inheritance	4.2 Defining Derived Classes 4.3 Single Inheritance 4.4 Making a Private Member Inherited 4.5 Multiple Inheritance 4.6 Multilevel Inheritance 4.7 Hybrid Inheritance 4.8 Virtual Base Class 4.9 Abstract Classes
	4c. Apply the concept of constructor in derived classes	4.10 Constructor in Derived Classes
Unit– V Pointers, Virtual functions and polymorphism	5a. Apply Pointer to objects	5.1 Pointers to objects, 5.2 Develop programs using pointers to objects
	5b. List applications of ‘this’ pointer	5.3 ‘this’Pointer
	5c. Define derived classes and virtual functions	5.4 Pointer to Derived Classes 5.5 Virtual Functions 5.6 Pointer to virtual Functions
Unit– VI Managing Console I/O Operations	6a. Apply various input and output formats on single set of data	6.1 Input and Output Streams 6.2 C++ Stream Classes 6.3 Unformatted and formatted I/O Operations
	6b. Develop programs using manipulators	6.4 Formatting with Manipulators

5. SUGGESTED SPECIFICATIONTABLE WITH HOURS & MARKS(THEORY)

Unit	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Principle of Object Oriented Programming	04	03	02	02	07
II	Functions in C++ and Working with objects	14	06	08	10	24
III	Constructor and Destructor	08	02	06	04	12
IV	Inheritance	08	04	05	05	14
V	Pointers, Virtual functions and polymorphism	02	02	02	02	06
VI	Managing Console I/O Operations	06	02	02	03	07
Total		**42**	**19**	**25**	**26**	**70**

Legends:R = Remember; U= Understand; A= Apply and above levels (Bloom's revised taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may slightly vary from above table.

6. SUGGESTED LIST OF EXERCISES/PRACTICAL

The practical/exercises should be properly designed and implemented with an attempt to develop different types of practical skills **(Course Outcomes in psychomotor and affective domain)** so that students are able to acquire the competencies (Programme Outcomes). Following is the list of practical exercises for guidance.

Note: Here only Course Outcomes in psychomotor domain are listed as practical/exercises. However, if these practical/exercises are completed appropriately, they would also lead to development of **Programme Outcomes/Course Outcomes in affective domain** as given in a common list at the beginning of curriculum document for this programme. Faculty should refer to that common list and should ensure that students also acquire those Programme Outcomes/Course Outcomes related to affective domain.

S. No.	UnitNo.	Exercises (Course Outcomes in Psychomotor Domain according to NBA Terminology)	Approx. Hrs. Required
1	I	Develop minimum 5 programs using control structures	2
2	I	Develop minimum 2 programs using arrays	2
3	I	Develop programs using reference variable, scope resolution operator, simple manipulators, and number data type.	4
4	II	Develop programs using call by reference and return by reference, default arguments, constant arguments, and	4

S. No.	UnitNo.	Exercises (Course Outcomes in Psychomotor Domain according to NBA Terminology)	Approx. Hrs. Required
		function overloading	
5	II	Define minimum 5 different classes such as student, distance, shape, employee, feet, time, data etc. with data member & member functions. Also Develop programs to test those classes functionality.	4
6	II	Develop Programs using array of objects and static member functions.	4
7	II	Develop programs to pass object as an argument and returning object.	4
8	III	Develop programs using various types of constructors and destructor.	4
9	III	Apply the concepts of constructors and destructors in the programs developed in unit-2 and test those programs.	4
10	IV	Develop programs using single, multilevel, multiple inheritance	6
11	IV	Develop programs using inheritance and constructors	4
12	V	Develop programs using pointer to derived classes	5
13	VI	Develop programs using unformatted i/o functions	5
14	VI	Develop programs using formatted i/o functions	4
		Total	**56**

Note:

- Develop i.e. write, debug, execute and test the program
- In tutorials - Students will write programs and in practical session -execute program

7. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:
Present seminar, develop mini projects, panel discussion, and develop a program with real life application examples on a particular topic.

8. SPECIAL INSTRUCTIONAL STRATEGIES (If Any)

The course activities include Lectures, Supervised Tutorials and Practical Exrecises as per teaching scheme. The programmes in C++ would be prepared during tutorials and would be executed during practicals sessions. Following needs attention:

i. Concepts may be introduced in lectures using charts/ppt.

ii. Quiz on various topics like class, polymorphism, inheritance etc.

iii. Role play by studnets for understanding concept of inheritance

iv. Problem solving will be done through tutorials.

9. SUGGESTED LEARNING RESOURCES

(A) List of Books

Sr. No.	Title of Books	Author	Publication
1	Object Oriented Programming in C++	Lafore, Robert	SAMS, 2012
2	Object Oriented Programming with C++	Balagurusamy, E.	McGrawHill, Delhi, 2012
3	Object Oriented Programming with C++ - second edition	Sahay, Sourav	Oxford, Delhi 2012
4	Mastering C++	Venugopal	Tata McGrawHill, Delhi, 2011
5	Programming in c++	Kamthane, Ashok	Pearson, New Delhi, 2012

(B) List of Major Equipment/Materials

i. Hardware: Computer System with minimum PIV processor (or equivalent) and 1 GB RAM.
ii. Software: Turbo C++/ Borland C++/ any other higher software

(C) List of Software/Learning Websites

i. C++ Fundamentals:http://www.oupinheonline.com
ii. C++ Tutorials: http://www.tutorialspoint.com/cplusplus/cpp_overview.htm
iii. Video tutorials of C++: http://nptel.iitm.ac.in/syllabus/syllabus.php?subjectId=106101006
iv. Learn C++ Programming: http://www.learncpp.com
v. Complete C++: http://www.cplusplus.com

10. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

- **Prof. R. M Shaikh**, Head Deapartment of Computer Engineering, KD Polytechnic,Patan.
- **Prof. K. N. Raval,** Head Deapartment of Computer Engineering, R C Technical Institute, Ahmedabad
- **Prof. (Ms.) M. P. Mehta** Sr. Lecturer Deapartment of Computer Engineering, K.D.Polytechnic, Patan

Coordinator and Faculty Members from NITTTR Bhopal

- **Dr. Shailendra Singh,** Professor & Head Department of Computer Engineering and Applications
- **Dr. K. J. Mathai,** Associate Professor Department of Computer Engineering and Applications

GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

Course Curriculum

DATABASE MANAGEMENT SYSTEM
(Code: 3330703)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering	3rd Semester

1. RATIONALE

The aim of this subject is to get broad understanding of the basic concepts of database management system in particular relational database system. The students will also develop the skills to design database system and develop application programs to manage & retrieve data from different perspective using Structured Query Language (SQL) in ORACLE.

2. COMPETENCY (Programme Outcome according to NBA Terminology):

The course should be taught and implemented with the aim to develop various types of skills so that students are able to acquire following competency:

- **Design, Develop and manage databases for simple applications using Structured Query Language (SQL) in ORACLE**.

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	150
3	2	2	7	70	30	20	30	

Legends: L -Lecture; **T** -Tutorial/Teacher Guided Student Activity; **P** -Practical; **C** -Credit;**ESE**-End Semester Examination; **PA** -Progressive Assessment

4. COURSE DETAILS

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit – I Introduction to Database System	1a. Differentiate the terms: Data, Information, Records, Fields, Metadata, Data warehouse, Data dictionary	1.1 Concepts and Definitions database and database systems and database environment 1.2 Data, Information, Data Item or Fields, Records, Files, Metadata, System Catalog, Data Warehouse, Data dictionary and it's components
	1b. List Functions of DA and DBA	1.3 Data Administrator (DA) and Database Administrator (DBA) 1.4 Functions and Responsibilities of DBAs
	1c. Compare File oriented approach and Database approach	1.5 Advantage and Disadvantages of File-oriented system 1.6 Advantage and disadvantages of DBMS, File oriented System versus database system
Unit– II Database System Architecture	2a. Define Schemas, Sub-schemas and instances	2.1 Schemas, Sub-schemas, and Instances
	2b. Explain Three-level ANSI SPARC database Architecture	2.2 Three-level ANSI SPARC Database Architecture: Internal Level, Conceptual Level, External Level, 2.3 Advantages of three-tier Architecture
	2c. Differentiate between physical and Logical Data Independence	2.4 Data Independence: Physical Data Independence, Logical Data Independence
	2d. Analyze Conceptual, Internal and External Mapping	2.5 Mappings: Conceptual / Internal Mapping,External / Conceptual Mapping
	2e. Explain the Components and Functions of DBMS	2.6 Structure Components, and Functions of DBMS: Structure of DBMS, Execution Steps of a DBMS, Components of a DBMS, Function and Services of DBMS
	2f. Explain various Data Models	2.7 Data Models: Record-based Data Models, Object based Data Models, Physical Data Models, Hierarchical Data Model, Network Data Model, Relation Data Model, Entity – Relationship (E-R) Data Model, Object – oriented Data Model, Comparison between Data Models
	2g. Explain various types of Database systems	2.8 Types of Database System: Centralized Database System, Parallel Database System, Client / Server Database System, Distributed Database System

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit– III Introduction to Structured Query Language (SQL)	3a. 3a. List data types in DBMS	3.1 Data types
	3b. 3b. Perform Data Definition Language (DDL) Commands for creating tables	3.2 Database language. Data Definition Language (DDL): CREATE,ALTER,TRUNCATE, DROP
	3c. 3c. Perform Data Manipulation Language (DML) commands for managing tables	3.3 Database language. Data Manipulation Language(DML) : INSERT,SELECT,UPDATE,DELETE
	3d. 3d. Execute various SQL operators and Functions	3.3 Operators Arithmetic, Comparison, Logical 3.4 SQL functions- Single row function i. Single row function. ii. Date functions (add-months,months-between, round,truncate, greatest, new-time). iii. Numeric Functions (abs, ceil, cos, cosh, exp, floor, power, mod, round, trunc, sqrt) iv. Character Fucntions (initcap, lower, upper, ltrim, rtrim, translate, replace, substring) v. Conversion Functions (to-char, to-date, to-number) vi. Miscellaneous functions (uid, user, nvl, vsize) vii. Group functions : Avg, Min,Max,Sum,Count, Decode
	3e. 3e. Perform queries on 'Group by', 'Having' and 'Order by' clause	3.5 Group by, Having and Order by clause
Unit– IV Relational Algebra and implementation using SQL	4a. Explain Relational Algebra and its notations	4.1 Structure of Relational Database 4.2 Domain 4.3 Keys of Relations
	4b. Derive the information using operations of Relational Algebra 4c. Implement set operations using SQL	4.4 Relational Algebra : Selection Operation, Projection Operation, Joining Operation, Outer join Operation, Union Operation, Difference Operation, Intersection Operation, Cartesian Product Operation, Division Operation, Examples of queries in Relation Algebraic using symbols 4.5 Implementing Relational Algebra using SQL 4.6 Set operators: Union, union all, Intersect, Minus

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
	4d. Implement 'Joins'	4.7 Joins: Simple, Equi-join, Non-equi, Self-Joins, Outer-joins.
	4e. Perform other types of queries	4.8 Sub queries Multiple, Correlated
	4f. Create report using formatting commands	4.9 Reports: Advanced formatting, Break on , Order of column in break on, Title, btitle and formatting commands, Break on row. 4.10 Adding views
Unit– V Database Integrity Constraints	5a. Explain with examples Domain Integrity and Entity Integrity constraint	5.1 Domain Integrity constraints:Not null, Check 5.2 Entity Integrity constraints:Unique, Primarykey.
	5b. Explain with examples Referential Integrity constraints	5.3 Referential integrity constaints: Foreign key, referenced key, on delete cascade
Unit– VI Entity Relational Model	6a. Explain E – R concepts-Entity, Relationship, Attributes	6.1 Basic Entity – Relationship Concepts: Entities, Relationship, Attributes 6.2 E – R Diagram symbols
	6b. Convert E – R model into Relational model	6.3 Conversion of Entity – Relationship Model into Relations
	6c. Solve problems with E – R Models	6.4 Problems with Enitty – Relationship Models
	6d. Explain Specialisation and Generalisation concepts of EER Model	6.5 Concepts : Specialisation and Generalisation

5. SUGGESTED SPECIFICATIONTABLE WITH HOURS & MARKS(THEORY)

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total
I.	Introduction to Database System	6	4	4	0	8
II.	Database System Architecture	7	4	6	2	12
III.	Introduction to Structured Query Language (SQL)	8	2	4	10	16
IV.	Relational Algebra and implementation using SQL	10	2	6	8	16
V.	Database Integrity Constraints	5	4	4	2	10
VI.	Entity Relational Model	6	2	2	4	8
	Total	**42**	**18**	**26**	**26**	**70**

Legends: R = Remember; U= Understand; A= Apply and above levels (Bloom's revised taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may slightly vary from above table.

6. SUGGESTED LIST OF PRACTICALS

The practical/exercises should be properly designed and implemented with an attempt to develop different types of practical skills **(Course Outcomes in psychomotor and affective domain)** so that students are able to acquire the competencies (Programme Outcomes). Following is the list of practical exercises for guidance.

Note: Here only Course Outcomes in psychomotor domain are listed as practical/exercises. However, if these practical/exercises are completed appropriately, they would also lead to development of **Programme Outcomes/Course Outcomes in affective domain** as given in a common list at the beginning of curriculum document for this programme. Faculty should refer to that common list and should ensure that students also acquire those Programme Outcomes/Course Outcomes related to affective domain.

S. No.	Unit No.	Practicals (Course Outcomes in Psychomotor Domain according to NBA Terminology)	Approx Hrs.
1	III	Implement SQL queries to perform various DDL Commands. (Create minimum 5 tables with different datatypes and operateupon them)	4
2	III	Implement SQL queries to perform various DML Commands. (Insert minimum 10 rows using different insert methods, edit and remove data using update and delete commands)	4
3	III	Retrieve data using SELECT command and various SQL operators.	4
4	III	Implement SQL queries using Date functions like add-months,months-between, round, nextday, truncate, greatest, new-time etc	4
5	III	Implement SQL queries using Numeric functions like abs, ceil, cos, cosh, exp, floor, power, mod, round, trunc, sqrt etc.	6
6	III	Implement SQL queries using Character Functionslikeinitcap, lower, upper, ltrim, rtrim, translate, replace, substring etc.	6
7	III	Implement SQL queries using Conversion Functions like to-char, to-date, to-number and Miscellaneous functions like uid, user, nvl, vsize etc.	6
8	III	Implement SQL queries using Group functions like Avg, Min,Max,Sum,Count, Decode etc.	6
9	III	Implement SQL queries using Group by, Having and Order by clause	6
10	IV	Implement SQL queries using Set operators like Union, union all, Intersect, Minus etc.	6
11	IV	Retrieve data spread across various tables or same table using various Joins.	6
12	IV	Retrieve data from multiple tables using Sub queries (Multiple, Correlated) (write minimum 3 level sub query)	6
13	IV	Tabulate formatted output using various report commands like Break on Title, btitleetc.	6
		TOTAL	**70**

Note: In tutorials - Students will write programs and in practical session -execute program

7. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:

i. Prepare seminar presentations explaining the organization of database in various live systems like banking, insurance, online booking etc.
ii. Mini projects such as: Prepare charts for database architecture, E – R Model, Relational algebra etc.

8. SPECIAL INSTRUCTIONAL STRATEGIES (If Any)

The course activities include Lectures, Supervised Tutorials and Practical Exrecises as per teaching scheme. The programmes would be prepared during tutorials and would be executed during practicals sessions

9. SUGGESTEDLEARNING RESOURCES

(A) List of Books

S. No.	Title of Books	Author	Publication and Year
1	Database Systems Concepts, design and Applications 2/e	Singh, S. K.	PearsonEducation, New Delhi, 2011
2	SQL/ PL/SQL	Bayross, Ivan	BPB, New Delhi, 2010.
3	An Introduction to Database Systems	Date, C. J.	PearsonEducation, New Delhi,2006
4	Database System Concepts,	Korth, Henry	McGrawHill, Delhi, 2011
5	Introduction to Database Systems	ITL ESL.	Pearson Education, New Delhi, 2010

B. List of Major Equipment/Materials

i. Hardware: Computer Systems with minimum PIV processor (or equivalent) and 1 GB RAM.
ii. Software: SQL/PLSQL supporting software. (e.g. Oracle, SQL Server, MySQL)

C List of Software/Learning Websites

i. DBMS:http://nptel.iitm.ac.in/video.php?subjectId=106106093
ii. SQL Plus Tutorial: http://holowczak.com/oracle-sqlplus-tutorial/
iii. Database Tutorials:http://www.roseindia.net/programming-tutorial/Database-Tutorials
iv. SQL Basic Concepts: http://www.w3schools.com/sql/
v. SQL Tutorial : http://beginner-sql-tutorial.com/sql.htm

10. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

- **Prof. R. M Shaikh** Head Department of Computer Engg., KD Polytechnic Patan
- **Prof. K. N. Raval** Head Department of Computer Engg., RCTI, Ahmedabad
- **Prof.S. D. Shah,** Lecturer Department of Computer Engg.,RCTI, Ahmedabad

Coordinator and Faculty Members from NITTTR Bhopal

- **Dr.Shailendra Singh**, Professor & Head, Dept. of Computer Engineering and Applications
- **Dr. K. J. Mathai**, Associate Professor, Dept. of Computer Engineering and Applications

GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

Course Curriculum

DATA STRUCTURES
(Code: 3330704)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering, Information Technology	3rd Semester

1. RATIONALE

Development of application systems and software that use underlying architecture of machines efficiently and effectively requires the ability to use and manipulate various types of Data Structures and other constructs. This being a fundamental ability which is language neutral, yet requires use of a language for its implementation. This is a basic course which goes along with other programming courses to develop an integrated ability to efficient software development, hence this course is very important for computer and IT diploma engineers.

2. COMPETENCY (Programme Outcome according to NBA Terminology):

The course content should be taught and implemented with the aim to develop various types of skills so that students are able to acquire following competency:

- **Implement various types of algorithms using Data Structures.**

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	
3	2	2	7	70	30	20	30	150

Legends: L -Lecture; **T** -Tutorial/Teacher Guided Student Activity; **P** -Practical; **C** - Credit;**ESE**-End Semester Examination; **PA** -Progressive Assessment

4. COURSE DETAILS

Unit	MajorLearning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit – I Basic Concepts of Data Structures	1a. Represent the data in relevant memory	1.1 Data Structure Basic Concepts 1.2 Types of data structures
	1b. Differentiate primitive and non-primitive data structures	1.3 Primitive and non-primitive data structures
	1c. List key features of an algorithm	1.4 Introduction to Algorithms 1.5 Key features of an algorithm
	1d. Define time complexity and space complexity	1.6 Analysis Terms (for the definitions purpose only) : a. Time Complexity b. Space Complexity c. Asymptotic Notations ,Big 'O', Notation , Best case Time Complexity, Average case Time Complexity, Worst case Time Complexity
	1e. Design and Implement programs to represent array in row major and column major order	1.7 Array : i. Row Major Arrays ii. Column Major Arrays 1.8 Overview of various array operations.
	1f. Design and Implement search algorithms	1.9 Searching an element into an array – i. Linear Search ii. Binary Search
Unit– II Strings	2a. Create strings	2.1 String representation : Reading and Writing Strings
	2b. Develop algorithms toimplement various operations on string	2.2 String operations : Finding length of a string, Converting Characters of a string into upper case and lower case, Concatenation of two strings to form a new string, Appending, Reversing a string, Copying a string, Comparing strings, Insertion, Substring, Deletion
Unit– III Stack and Queues	3a. Define linear and non-linear data structures and develop algorithms to push an element into stack, pop an element from the stack.	3.1 Linear and Non-Linear Data Structures 3.2 Stack : Array representation of Stack, PUSH POP Operations on Stack, Implementation of Stack, Application of Stack, Infix, Prefix and Postfix Forms of Expressions, Recursive Functions (factorial, greatest common divisor, Fibonacci series)

Unit	MajorLearning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
	3b. Implement Queue with various operations like insert delete	3.3 Queue: Array representation of Queue, Operations on Queue, Implementation of Queue, Limitation of Single Queue
	3c. Differentiate circular and simple queue	3.4 Concepts of Circular Queue 3.5 Application of queue 3.6 Difference circular queue and simple queue
Unit– IV Linked List	4a. Define linked list	4.1 Pointers Revision 4.2 Revision of Structure 4.3 Revision of structure using pointers 4.4 Dynamic Memory Allocation 4.5 Linked list Presentation 4.6 Types of Linked List
	4b. Develop algorithms to insert node from front, to the end, at any position, delete element, insert into sorted list, delete node from singly linked list	4.7 Basic operations on singly linked list : Insertion of a new node in the beginning of the list, at the end of the list, after a given node, before a given node, in sorted linked list, Deleting the first and last node from a linked list, Searching a node in Linked List, Count the number of nodes in linked list
	4c. Distinguish circular linked list and singly linked list	4.8 Concepts of circular linked list 4.9 Difference between circular linked list and singly linked list
	4d. Develop algorithms to insert node from front, to the end, at any position, delete node from doubly linked list	4.10 Basic operations on Doubly linked list : Insertion of a new node in the beginning of the list, at the end of the list, after a given node, before a given node. Deleting the first and last node from a linked list, Searching a node in Linked List, Count the number of nodes in linked list
	4e. List the applications of linked list	4.11 Applications of linked list
Unit– V Sorting and Hashing	5a. Arrange data in ascending and descending orders using appropriate sorting algorithm	5.1.Sorting Methods : a. Bubble Sort, b. Selection Sort, c. Quick Sort, d. Insertion Sort, e. Merge Sort, f. Radix Sort
	5b. Apply various hashing techniques	5.2.Hashing Concepts 5.3.Hash functions : Division Method, Middle Square Method, Folding Method,

Unit	MajorLearning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
	5c. Apply collision resolution techniques	5.4.Collision in Hashing 5.5.Collision resolution techniques: Linear Probing
Unit– VI Trees	6a Define non-linear data structure	6.1 Non-linear data structures
	6b Develop algorithms to manipulate tree	6.2 Binary trees : Complete Binary Tree, Basic Terms: level number, degree, in-degree and out-degree, leaf node, similar binary trees, copies of binary trees, directed edge, path, depth, General Tree, Conversion of General Tree to Binary Tree, Forest.
	6c Implement various tree manipulation algorithms	6.3 Binary Search Tree : Insertion of a node in binary tree, Deletion of a node in binary tree, Searching a node in binary tree 6.4 Tree Traversal : Inorder, Preorder, Postorder
	6d List applications of tree	6.5Applications of binary tree

5. SUGGESTED SPECIFICATIONTABLE WITH HOURS & MARKS(THEORY)

Unit	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Basic Concepts of Data Structures	4	4	3	0	7
II	Strings	4	2	2	3	7
III	Stack and Queues	8	2	6	6	14
IV	Linked List	10	4	8	2	14
V	Sorting and Hashing	8	2	6	6	14
VI	Trees	8	4	4	6	14
Total		42	18	29	23	70

Legends: R = Remember; U= Understand; A= Apply and above levels (Bloom's revised taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may slightly vary from above table.

6. SUGGESTED LIST OF EXERCISES/PRACTICAL

The practical/exercises should be properly designed and implemented with an attempt to develop different types of practical skills **(Course Outcomes in psychomotor and affective domain)** so that students are able to acquire the competencies (Programme Outcomes). Following is the list of practical exercises for guidance.

Note: Here only Course Outcomes in psychomotor domain are listed as practical/exercises. However, if these practical/exercises are completed appropriately, they would also lead to development of **Programme Outcomes/Course Outcomes in affective domain** as given in a common list at the beginning of curriculum document for this programme. Faculty should refer to that common list and should ensure that students also acquire those Programme Outcomes/Course Outcomes related to affective domain.

Sr. No.	Unit No.	Practical Exercises (Course Outcomes in Psychomotor Domain according to NBA Terminology)	Approx. Hrs. Required
1	I	Define various terms such as algorithm, various approaches to design an algorithm, time complexity, space complexity, big 'o' notation, best case, average case and worst case time complexity etc. Develop simple program using pointer to a structure	2
2	I	Implement array using row major order and column major order	2
3	I	Implement Sequential search algorithms	2
4	I	Implement Binary search algorithms	2
5	II	Implement various string algorithms	2
6	III	Implement push and pop algorithms of stack using array	2
7	III	Implement recursive functions	2
8	III	Implement insert, delete algorithms of queue using array	2
9	III	Implement insert, delete algorithms of circular queue	2
10	IV	Implement simple structure programs using pointers	2
11	IV	Implement insertion of node in the beginning of the list and at the end of list in singly linked list	4
12	IV	Implement insertion of node in sorted linked list	2
13	IV	Implement insertion of node at any position in liked list	2
14	IV	Implement searching of a node, delete a node and counting no of node algorithms in singly linked list	4
15	IV	Implement insertion of node in the beginning and at the end of doubly linked list	2
16	IV	Implement insertion of node at any position in doubly liked list	2
17	IV	Implement searching of a node, delete a node and counting no of node algorithms in doubly linked list	4
18	V	Implement Bubble sort, Selection sort algorithms	2
19	V	Implement Quick Sort	2
20	V	Implement Insertion sort, Shell sort algorithm	2

21	V	Implement Merge Sort algorithm	2
22	V	Solve hash table example using division method, method square method, folding method (paper work only)	2
23	VI	Implement construction of binary search tree	2
24	VI	Implement inorder, preorder and postorder traversal methods in binary search tree	2
25	VI	Implement searching algorithm in binary search tree	2
		Total	**56**

Note: In tutorials - Students will write programs and in practical session -execute program

7. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:
Prepare seminars on various topics like stack, queue, singly linked list algorithms, doubly linked list, sorting algorithms, tree algorithms etc.
Prepare diagrams/chart to display various sorting methods, string representation; data representations etc.
Develop a program with real life application example of particular topic.
Compare various sorting and searching methods. Prepare hash tables using given methods.

8. SPECIAL INSTRUCTIONAL STRATEGIES (If Any)

The course activities include Lectures, Supervised Tutorials and Practical Exrecises as per teaching scheme. The programmes in C++ would be prepared during tutorials and would be executed during practicals sessions. Following needs attention:

i. Concepts will be introduced in lectures using charts.
ii. Role play by students for explaining concepts of array, stack, queue and sorting
iii. Problem solving will be done through tutorials.
iv. Practical work will be through laboratory sessions.
v. Debate/Group Discussions for comparision of searching and sorting methods
vi. Solving Puzzels

9. SUGGESTED LEARNING RESOURCES

A. List of Books

S. No.	Title of Books	Author	Publication
1	Data and File Structures using C	Thareja,Reema	Oxford University Press New Delhi 2011
2	Data Structures using C	ISRD Group	McGraw Hill, New Delhi, 2013
3	Data Structures	Chitra, A Rajan, P T	Tata McGraw Hill, New delhi, 2009
4	Classic Data Structures	Samanta, D.	PHI Learning, New Delhi

B. List of Major Equipment/Materials with broad specifications

i. Hardware: Computer System with minimum PIV processor (or equivalent) and minimum 1 GB MB RAM.

ii. Software: Turbo C++/ Borland C++/ any other higher software

C. List of Software/Learning Websites

i. Turbo C/C++ or Borland C/C++ or any software that support c/c++ compiler

ii. Data Structure Introduction:http://nptel.iitm.ac.in/video.php?subjectId=106102064

iii. Data Structure Tutorial: http://www.roseindia.net/tutorial/datastructure/

iv. Data Structure Fundamentals: http://www.cprograms.in/index.htm

v. Data structure videohttp://www.youtube.com/watch?v=tORLeHHtazM

10. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

- **Dr. P.P.Kotak,** Head Deptt. of Computer Engineering, AVPTI, Rajkot
- **Prof. K. N. Raval,** Head Deptt. of Computer Engg., RCTI, Ahmedabad
- **Prof. Rahul B. Pancholi**, Lecturer, Deptt. of Computer Engg., L. J. Ahmedabad
- **Prof.(Mrs.) R.K.Vaghela,** Deptt. of Computer Engg., RCTI , Ahmedabad

Coordinator and Faculty Members from NITTTR Bhopal

- **Dr. Shailendra Singh**, Professor & Head Deptt. of Computer Engg. and Applications
- **Dr. PriyankaTripathi,** Associate Professor, Deptt. of Computer Engg. and Applications

GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

Course Curriculum

MICROPROCESSOR AND ASSEMBLY LANGUAGE PROGRAMMING (Code: 3330705)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering	3rd Semester

1. RATIONALE

The microprocessor is challenging and very dynamic field. This course explores architecture of a microprocessor and its programming in assembly language. The student will be able to apply logics to various given problems and develop programs using assembly language construct that would help them to develop real time microprocessor based application programs.

2. COMPETENCY (Programme Outcome according to NBA Terminology):

The course content should be taught and implemented with the aim to develop different types of skills so that students are able to acquire following competency:

- Develop code, debug, test and execute various assembly language programs using 8085 instruction set.

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	150
3	0	2	5	70	30	20	30	

Legends: L - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** - Practical; **C** - Credit; **ESE** - End Semester Examination; **PA** - Progressive Assessment

4. COURSE DETAILS

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Unit – I Introduction of Microprocessor	1a. Classify types of microprocessor.	1.1 Evolution of microprocessor and it's types
	1b. Define various buses and their functions	1.2 Microprocessor Bus organization : Data Bus, Address Bus and Control Bus
	1c. List the operations of microprocessor	1.3 Operations of microprocessor: internal data manipulation, microprocessor initiated and peripheral or external initiated
	1d. Draw and Explain microprocessor architecture.	1.4 Pin diagram and block diagram of 8085 1.5 Architecture of 8085 1.6 Internal registers organization of 8085 1.7 Limitations of 8085
Unit – II Instruction Cycle and Timing Diagram	2a. Explain the bus timing diagram	2.1 8085 machine cycle and bus timings to fetch, decode and execute instruction from memory
	2b. Draw machine cycle with timing diagram	2.2 Memory read and write 2.3 Input/Output read and write cycle with timing diagram
Unit – III 8085 Instruction set	3a. Interpret machine language format, addressing modes and operations with various types of instructions	3.1 Machine language instruction format : Single byte, two byte, three byte instructions 3.2 Various addressing modes 3.3 Data transfer operation and instruction 3.4 Arithmetic operation and instruction 3.5 Logical operation and instruction 3.6 Branch operation and instruction 3.7 Stack operation and instruction 3.8 Input/Output and machine control operation and instruction
	3b. Develop simple programs	3.9 Simple programs with 8085 instruction
Unit – IV Programming Techniques of 8085	4a. Develop programs using control structures	4.1. Looping, Counting and indexing 4.2. Counter and Timing delays
	4b. Implement subroutines, procedures.	4.3. Stack and subroutine basic concepts 4.4. Procedure and macro
Unit – V 8085 Interrupts	5a. Classify interrupts and list its applications	5.1 Interrupts and its need 5.2 classification of interrupts 5.3 8085 interrupts: software, hard ware, and priorities of interrupts 5.4 8085 vectored interrupts:TRAP,INTR 5.5 RST 7.5, RST 6.5, RST 5.5
Unit – VI Introduction to Advanced	6a. Draw block diagram and 6b. Explain 8086 microprocessor architecture.	6.1. Block diagram - Architecture of 8086 6.2. Register organization of 8086

Unit	Major Learning Outcomes (Course Outcomes in Cognitive Domain according to NBA terminology)	Topics and Sub-topics
Microprocessor	6c. Explain instruction pipelining and memory segmentation	6.3. Concepts of Instruction pipelining 6.4. Memory segmentation 6.5. Memory address generation
	6d. Differentiate maximum and minimum mode operation with diagram	6.6. Minimum and Maximum mode operation and diagram

5. SUGGESTED SPECIFICATION TABLE WITH HOURS & MARKS (THEORY)

Unit	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Introduction of Basic Microprocessor	6	7	7	0	14
II	Instruction Cycle and Timing Diagram	4	0	7	0	07
III	8085 Instruction set	12	0	7	7	14
IV	Programming Techniques of 8085	8	0	0	14	14
V	8085 Interrupts	4	0	7	0	07
VI	Introduction to Advanced Microprocessor	8	7	7	0	14
Total		**42**	**14**	**35**	**21**	**70**

Legends: R = Remember; U= Understand; A= Apply and above levels (Bloom's revised taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

6. SUGGESTED LIST OF EXERCISES/PRACTICAL

The practical/exercises should be properly designed and implemented with an attempt to develop different types of practical skills (Course Outcomes in psychomotor and affective domain) so that students are able to acquire the competencies (Programme Outcomes). Following is the list of practical exercises for guidance.

Note: Here only Course Outcomes in psychomotor domain are listed as practical/exercises. However, if these practical/exercises are completed appropriately, they would also lead to development of Programme Outcomes/Course Outcomes in affective domain as given in a common list at the beginning of curriculum document for this programme. Faculty should refer to that common list and should ensure that students also acquire those Programme Outcomes/Course Outcomes related to affective domain.

Sr. No.	Unit No	Practical/Exercise (Course Outcomes in Psychomotor Domain according to NBA Terminology)	Apprx. Hrs. Required
1	I	Demonstrate of kit/simulator of 8085	2
2	III	Implement program to perform arithmetic operations(Add, subtract, multiply and divide) on signed and unsigned two 8 bit numbers.	6
3	III	Implement a program to mask the lower four bits of content of the memory location.	2
4	III	Implement a program to set higher four bits of content of the memory location to 1.	2
5	III	Implement a program to perform Exclusive OR of two numbers.	2
6	III	Implement a program to exchange the content of two memory locations.	2
7	III	Implement program to add/subtract 16 bit numbers	2
8	III	Implement program to copy content of one memory location to another memory location.	2
9	III	Implement a program to check whether given no is odd or even.	2
10	IV	Implement a program to compare two numbers.	2
11	IV	Implement a program to sum integers from 0 to 9.	2
12	IV	Implement a program to count no of zero value in given block of data.	2
13	IV	Implement program to add an array of data	2
14	IV	Implement program to find the smallest and the largest number from an array of n number	2
15	IV	Implement a program to move/copy block of memory to another given location and length of block is given in specific memory location.	2
16	IV	Implement a program to calculate the length of a string given starting and ending location.	2
17	IV	Implement a program to reverse a string using stack operations and stored in same memory area.	2
18	IV	Implement a program to find minimum and maximum from block data using subroutine provided length of block stored at memory location.	2
19	IV	Implement program to count no of negative values in given block of data.	2
		Total	**42**

7. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:

a. Develop programs related with unit wise topics in computer laboratory.
b. Develop any module of to be useful in real life application.
c. Multimedia presentation of module developed by students.

8. INSTRUCTIONAL STRATEGIES

Concepts should be explained thoroughly in theory sessions and should be implemented in laboratory appropriately along with the problem solving. Concept should be developed by giving problems to students as assignments.

9. SUGGESTED LEARNING RESOURCES

A. List of Books

Sr. No.	Title of Books	Author	Publication and year
1	8085 Microprocessor and its application	Kani, A. Nagoor	Mc Graw Hill 2013
2	Microprocessor & Interfacing Programming and hardware Revised second edition	Hall, Douglas	Tata Mc Graw Hill 2006
3	Microprocessor Architecture, Programming, and Application with the 8085 (5th edition)	Gaonkar,Ramesh	Penram Mumbai 2009 edition or latest

B. List of Major Equipment/Materials

i. 8085 microprocessor kits
ii. Computer Systems with minimum PIII processor (or equivalent) and 512 MB RAM.
iii. Multimedia Projector

C. List of Software/Learning Websites

i. Simulator such as : http://8085simulator.codeplex.com/ http://gnusim8085.org/ or its equivalent
ii. Latest processor configuration : http://www.intel.com/pressroom/kits/quickreffam.htm
iii. Intel 8085 microprocessor architecture: http://www.cpu-world.com/Arch/8085.html
iv. 8085 sample programs :http://www.8085projects.info/page/free-programs-for-8085-microprocessor.aspx
v. 8085 ppts: http://www.slideshare.net/shashank03/assembly-language-programming-of-8085
vi. Intel 8086 microprocessor architecture: http://www.cpu-world.com/Arch/8086.html

10. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

- **Prof. R. M Shaikh**, Head Deptt. of Computer Engg. K D Polytechnic, Patan.
- **Prof.(Ms.) M. P. Mehta**, Sr. Lecturer Deptt. of Computer Engg. K. D. Polytechnic, Patan
- **Prof. S. D. Shah**, Lecturer Deptt. of Computer Engg., RCTI, Ahmedabad

Coordinator and Faculty Members from NITTTR Bhopal

- **Dr. Shailendra Singh**, Professor & Head Dept. of Computer Engineering and Applications
- **Dr. M.A. Rizvi,** Associate Professor, Deptt. of Computer Engineering and Applications.