GOVERNMENT DEGREE COLLEGE FOR WOMEN
(AUTONOMOUS)
BEGUMPET, HYDERABAD

DEPARTMENT OF COMPUTER APPLICATIONS

PROGRAM OUTCOME:

1. Students will establish themselves as effective professionals by solvingreal problems through the use of computer science knowledge and withattention to team work, effective communication, critical thinking andproblem solving skills.

2. Students will develop professional skills thatprepare them for immediate employment and for life-long learning inadvancedareasofcomputerscience andrelatedfields.

3. An ability to function effectively on teams to accomplish a common goal.

4. An understanding of professional, ethical, legal, security and social issues and responsibilities. An ability to communicate effectively with a wide range of audiences.

5. An ability to analyze the local and global impact of computing on individuals, organizations, and society.

6. An ability to use current techniques, skills, and tools necessary for computing practice.

7. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

PROGRAMME SPECIFIC OUTCOME

1. The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, webdesign, application program, database , graphics and networking forefficientdesignofcomputer-basedsystemsofvaryingcomplexity.

2. Ability to provide socially acceptable technical solutions to complex computer science engineering problems with the application of modern and appropriate techniques for sustainable development relevant to professional engineering practice.

3. Ability to apply the knowledge of ethical and management principles required to work in a team as well as to lead a team. Ability to comprehend and write effective project reports in multidisciplinary environment in the context of changing technologies.

DEPARTMENT OF COMPUTER SCIENCE & APPLICATIONS
ALLOCATION OF CREDITS

S.no	Semester	Nature of Module	Name of Module	Hours	Max. Marks	Credits
1	Paper-I(A)	Core -1	PROGRAMMING IN C	4	100	3
2	Practicals		PROGRAMMING IN C	3	50	1
3	Paper-I(B)	Core - 2	PROGRAMMING IN C++	4	100	3
4	Practicals		PROGRAMMING IN C++	3	50	1
5	Paper-II(A)	Core - 3	OOPs WITH C++	4	100	3
6	Practicals		OOPs WITH C++	3	50	1
7	Paper-II(B)	Core - 4	RDBMS	4	100	3
8	Practicals		RDBMS	3	50	1
9	Paper-III(A)	Core- 5	JAVA PROGRAMMING-I	4	100	3
10	Practicals		JAVA PROGRAMMING-I	3	50	1
11	Paper-IV(A)	Advanced –1 Elective – 1 or Elective - 2	WEB TECHNOLOGIES-I Or OPERATING SYSTEM-I	4	100	02*
12	Practicals		WEB TECHNOLOGIES-I Or OPERATING SYSTEM-I	3	50	1
13	Paper-III(B)	Core- 6	JAVA PROGRAMMING-II	4	100	3

14	Practicals		JAVA PROGRAMMING-II	3	50	1
15	Paper-IV(B)	Advanced –2 Elective – 1 or Elective - 2	WEB TECHNOLOGIES-II Or OPERATING SYSTEM-II	4	100	02*
16	Practicals		WEB TECHNOLOGIES-II Or OPERATING SYSTEM-II	4	50	1
17	Trans Disciplinary		Web Designing (Sem-III & IV)	1	50	2

P.P.W: 6(4Th+2Pr)
Credits: 4Th+1Pr

COURSE CODE:CA101

Course Outcome:

1. Able to implement the algorithms and draw flowcharts for solving Mathematical and Engineering problems.
2. Demonstrate an understanding of computer programming language concepts. To be able to develop C programs on Linux platform.
3. Ability to design and develop Computer programs, analyzes, and interprets the concept of pointers, declarations, initialization, operations on pointers and their usage.
4. Able to define data types and use them in simple data processing applications also he/she must be able to use the concept of array of structures.
5. Student must be able to define union and enumeration user defined data types. Develop confidence for self education and ability for life-long learning needed for Computer language.

Syllabus

Unit – I :
Computer Fundamentals: Introduction of Computers, Classification of Computers, Anatomy of a Computer, Memory Hierarchy, Introduction to OS, Operational Overview of a CPU.
Program Fundamentals: Generation and Classification of Programming Languages, Compiling, Interpreting, Loading, Linking of a Program, Developing Program, Software Development.
Algorithms: Definitions, Different Ways of Stating Algorithms (Step-form, Pseudo-code, Flowchart), Strategy for Designing Algorithms, Structured Programming Concept.
Basics of C: Overview of C, Developing Programs in C, Parts of Simple C Program, Structure of a C Program, Comments, Program Statements, C Tokens, Keywords, Identifiers, Data Types, Variables, Constants, Operators and Expressions, Expression Evaluation–precedence and associatively, Type Conversions.

Unit – II :
Input-Output: Non-formatted and Formatted Input and Output Functions, Escape Sequences,
Control Statements: Selection Statements – if, if-else, nested if, nested if-else, comma operator, conditional operator, switch; Iterative Statements – while, for, do-while; Special Control Statement – goto, break, continue, return, exit.
Arrays and Strings: One and Two Dimensional Arrays, Character Arrays, Functions from ctype.h, string.h.

Unit – III :

Functions: Concept of Function, Using Functions, Call-by-Value Vs Call-by-reference, Passing Arrays to Functions, Score of Variables, Storage Classes, Inline Functions, and Recursion.

Pointers: Introduction, Address of Operator (&), Pointer, Uses of Pointers, Arrays and Pointers, Pointers and Strings, Dynamic Memory Allocation.

Unit – IV :

User-defined Data Types: Declaring a Structure (Union) and its members, Initialization Structure (Union), Accessing members of a Structure (Union), Structures Vs Unions, Enumeration Types.

Files: Introduction, Using Files, Working with Text Files and Binary Files, Other File Management Functions. Text PradipDey, ManasGhosh, Computer Fundamentals and Programming in C (2e)

References:

1. Ivor Horton, Beginning C

2. Herbert Schildt, The Complete Reference C

3. Paul Deitel, Harvey Deitel, C How To Program

4. Byron S. Gottfried, Theory and Problems of Programming with C

5. Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language

6. B. A. Forouzan, R. F. Gilberg, A Structured Programming Approach Using

B.A (C/A) I Year
Subject: Computer Applications
Paper: I (A) Programming in C

Practical: 2 Hours/Week   **Credit: 1**

Course Outcome:

1. Know concepts in problem solving
2. To do programming in C language
3. To write diversified solutions using C language

Practical Question Bank

1. Write a program to find the largest two numbers using if and conditional operator.

2. Write a program to calculate arithmetic operations of two numbers using switch.

3. Write a program to print the reverse of a given number.

4. Write a program to print whether the given number is a prime or not.

5. Write a program to find largest and smallest elements in a given list of numbers.

6. Write a program to find the sum of two matrices.

7. Write a program to find the product of two matrices.

8. Write a program to print the reverse of a given string.

9. Write a program to find the factorial of a positive integer using iteration and recursion.

10. Write a program to find the GCD of two positive integers using iteration and recursion.

11. Write a program to demonstrate the call by value and the call by reference concepts.

12. Write a program to illustrate the use of Enumeration data type.

13. Write a program to illustrate the use of structure concept.

14. Write a program to illustrate the use of union concept.

15. Write a program to write content into a file and display contents of a file

16.   Write a program to copy content of one file into another file and display the content of new file.

P.P.W: 6 (4T+2P)
Credits: 5 (4T+1P)

COURSE CODE:CA201

Course Outcome:

1. To understand how C++ improves C with object-oriented features.
2. To learn how to write inline functions for efficiency and performance.
3. To learn the syntax and semantics of the C++ programming language.
4. To learn how to design C++ classes for code reuse.
5. To learn how to implement copy constructors and class member functions.
6. To understand the concept of data abstraction and encapsulation.
7. To learn how to overload functions and operators in C++.
8. To learn how containment and inheritance promote code reuse in C++.
9. To learn how inheritance and virtual functions implement dynamic binding with polymorphism.
10. To learn how to design and implement generic classes with C++ templates.
11. To learn how to use exception handling in C++ programs.

Syllabus

Unit – I:

Introduction to C++: Applications, Example Programs, Tokens, Data Types, Variables, Operators, Expressions, Control Structures, Arrays, Strings, Pointers.

Functions: Introduction, Prototype, Passing Data by Value, Reference Variables, Using Reference Variables as Parameters, Inline Functions, Default Arguments, Overloading Functions, Passing Arrays to Functions.

Unit – II:

Object Oriented Programming: Procedural Programming Vs Object-Oriented Programming, Terminology, Benefits, Languages, and Applications.

Classes: Introduction, Defining an Instance of a Class, Why Have Private Members? Separating Class Specification from Implementation, Inline Member Functions, Constructors, Passing

Arguments to Constructors, Destructors, Overloading Constructors, Private Member Functions, Instance and Static Members, Friends of Classes, Member wise Assignment, Copy Constructors, Operator Overloading.

Unit – III:

Inheritance: Introduction, Protected Members and Class Access, Base Class Access Specification, Constructors and Destructors in Base and Derived Classes, Redefining Base Class Functions, Polymorphism and Virtual Member Functions, Abstract Base Classes and Pure Virtual Functions, Multiple Inheritance.

C++ Streams: Stream Classes, Unformatted I/O Operations, Formatted I/O Operations.

Unit – IV:

Exceptions: Introduction, Throwing an Exception, Handling an Exception, Object-Oriented Exception Handling with Classes, Multiple Exceptions, Extracting Data from the Exception Class, Re-throwing an Exception.

Templates: Function Templates–Introduction, Function Templates with Multiple Type, Overloading with Function Templates, Class Templates – Introduction, Defining Objects of the Class Template, Class Templates and Inheritance.

Text Book- Tony Gaddis, Starting out with C++: from control structures through objects (7e)

References

1. B. Lippman, C++ Primer

2. Bruce Eckel, Thinking in C++

3. K.R. Venugopal, Mastering C++

4. Herbert Schildt, C++: The Complete Reference

5. BjarneStroustrup, The C++ Programming Language

6. SouravSahay, Object Oriented Programming with C++

Practical: 2 Hours/Week   **Credit: 1**

Course Outcome:

1. Creating simple programs using classes and objects in C++.
2. Implement Object Oriented Programming Concepts in C++.
3. Develop applications using stream I/O and file I/O.
4. Implement simple graphical user interfaces.
5. Implement Object Oriented Programs using templates and exceptional handling concepts.

Practical Question Bank

1. Write a program to print the sum of digits of a given number

2. Write a program to check whether the given number is Armstrong or not

3. Write a program to check whether the given string is Palindrome or not

4.   Write a program to read the student name, roll no, marks and display the same using class and object.

5. Write a program to find area of a rectangle, circle, and square using class and object.

6. Write a program to implement inline function inside and outside of a class for

a. Finding the area of a square

b. Finding the area of a cube

7. Write a program to implement friend function and friend class

8. Write a program to implement constructor and destructor with in a class.

9. Write a program to demonstrate hierarchical inheritance.

10. Write a program to demonstrate multiple inheritances.

11. Write a program to demonstrate the constructor overloading.

12. Write a program to demonstrate static polymorphism.

13. Write a program to demonstrate dynamic polymorphism.

14. Write a program to implement polymorphism using pure virtual functions.

B.A II
YEAR
Semester-III
Subject: Computer
Applications Paper-II (A) OOPs
with C++

P.P.W: 6(4Th+2Pr)
COURSE CODE:CA301   **Credits: 3Th+1Pr**

Course Outcome:

1. Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects.
2. Understand dynamic memory management techniques using pointers, constructors, destructors, etc
3. Describe the concept of function overloading, operator overloading, virtual functions and polymorphism.
4. Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming.
5. Demonstrate the use of various OOPs concepts with the help of programs.

Syllabus

UNIT – I

Basic Concepts of OOPs: Comparison of procedural programming and OOP. Advantages of OOP and OOP Languages, definition of Class, Objects, Inheritance. Definition Encapsulation, Operator over Loading and Dynamic Binding.
Over view of OOP using C++: Basic Program Construction and Program Statement, Class Declaration, comments and C++ Compilation.
Elements of C++ Language: Tokens and Identifiers, Variables and Constants, Data Types, operators.

UNIT – II

Control Statements: The if Statement, if –else Statement, Switch Statements.
LOOP Statements: For Loop and While, do-While Statements, break, Continue and goto Statements.

UNIT-III

Classes and Objects: Declaration of classes and Objects in C++, Objects as function arguments. Object from Function, Structures and Classes, Constructors and Destructors.

UNIT – IV

Inheritance: Derived classes and Base class, Public and private Inheritance, Multiple Inheritance, Introduction to polymorphism, Runtime Polymorphism,

Suggested Books:
Prescribed Books:

C++ Programming: E.Balaguruswamy
Object Oriented Programming with C++: E.Balaguruswamy

Reference Books:

Using C++: Rob McGRegor

B.A II Year Practical Question Bank
Subject: Computer Applications
Paper: II (A) OOPs with C++

Course Outcome:

1. Develop solutions for a range of problems using objects and classes.
2. Programs to demonstrate the implementation of constructors, destructors and operator overloading.
3. Apply fundamental algorithmic problems including type casting, inheritance, and polymorphism.
4. Understand generic programming, templates, file handling.

Practical Question Bank

1. Write a C++ program for the implementation of the concept classes and objects.

2. Write a C++ program for the implementation of the concept Encapsulation.

3. Write a C++ program for the implementation of the concept Inheritance.

4. Write a C++ program for the implementation of the concept of Polymorphism.

5. Write a C++ program for the implementation of the concept of operator overloading.

6. Write a C++ program for the implementation of the concept of Multiple Inheritance.

7. Write a C++ program using Friend function.

8. Write a C++ program to find the factorial of a number using classes.

9. Write a C++ program to find addition of two matrices.

10. Write a C++ program to reverse a number using classes.

11. Write a C++ program to find whether a given number is palindrome or not using classes.

12. Write a C++ program to print FiboNAACi series a number using classes.

13. Write a C++ program to find the greatest number among 2 numbers using classes.

14. Write a C++ program to find the given year is leap year or not using classes.

15. Write a C++ program to print the multiplication table .

Department Of Computer Science & Applications
Trans Disciplinary Elective
Semester-III
Subject: Web Designing
Paper-II (A)

COURSE CODE:CA302

No. of Lectures: 15 No. of Credits: 1

Course Outcome:

1. Be able to use the HTML programming language.
2. Resolves written HTML codes.
3. Runs the page he/she has designed using HTML codes.
4. Be able to use the Design Programs.
5. Uses Microsoft Expression Web 4 programme.
6. Designs site and page via Microsoft Expression Web 4 programme.
7. Uses the program Web Page Maker.
8.

Syllabus

UNIT-I:
Introduction to internet, applications of internet, Web Browser, various Web Browsers and Client-Server Techniques.

UNIT-II
Introduction to HTML:
Basics-Introduction to HTML (Hypertext markup language)-Structure of HTML-Formatting tags-Physical tags-Logical tags -Header tags.

UNIT-III
Creating tables:
Creating Tables-Adding Borders- Cell Padding- Cell Spacing- Spanning- Using Colors-Aligning Tables in web Pages- Using Images in tables- Nesting Tables.

B.A (C/A) II
YEAR
Semester-IV
Subject: Computer Applications
Paper- II (B) Relational Database Management Systems

COURSE CODE:CA401

Course Outcome:

1. Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
2. Design ER-models to represent simple database application scenarios
3. Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
4. Improve the database design by normalization.
5. Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

Syllabus

Unit – I
Introduction to Databases: Introduction, Traditional File-Based Systems, Database Approach, Advantages and Disadvantages of DBMSs, The Three-Level ANSI-SPARC Architecture, Database Languages, Data Models, Functions of a DBMS, Components of a DBMS.
Relational Model: Introduction, Terminology, Integrity Constraints, Views.
Unit – II
SQL: Introduction, Data Manipulation Language commands, Sorting Results, Using the SQL Aggregate Functions, Grouping Results, Sub-queries, ANY and ALL, Joins, EXISTS and NOT EXIST, Combining Result Tables.
SQL: The ISO SQL Data Types, Data Definition Language commands –Creating an Index, Removing an Index, Views–Creating a View, Removing a View, WITH CHECK OPTION, Advantages and Disadvantages of Views.
Unit – III
Advanced SQL: The SQL Programming Language–Declarations, Assignments, Control Statements, Exceptions, Cursors, Subprograms, Stored Procedures, Functions, and Packages, Triggers, Recursion.
Entity–Relationship Modeling: Entity Types, Relationship Types, Attributes Keys, Strong and Weak Entity Types, Problems with ER Models–Fan Traps, Chasm Traps.
Enhanced Entity–Relationship Modeling: Specialization/Generalization, Aggregation, and Composition.
Unit – IV
Functional–Dependencies: Anomalies, Partial Functional Dependency, Transitive Functional Dependency.
Normalization: The Purpose of Normalization, Data Redundancy, Functional Dependencies, 1NF, 2NF, 3NF, BCNF.

Transaction Management: Transaction Support–Properties of Transactions, Concurrency Control–The Need for Concurrency Control, Serializability and Recoverability, Locking Methods, Deadlock, Time Stamping Methods.

Text Books:

Text Thomas M. Connolly, Carolyn E. Begg, Database Systems–A Practical Approach to Design, Implementation, and Management (6e)
Sharon Allen, Evan Terry, Beginning Relational Data Modeling

Reference

Jeffrey A. Hoffer, V. Ramesh, HeikkiTopi, Modern Database Management Raghu Ramakrishnan, Johannes Gehrke, Database Management Systems RamezElmasri, Shamkant B. Navathe, Fundamentals of Database Systems Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts
Carlos Coronel, Steven Morris, Peter Rob, Database Systems: Design, Implementation, and Management

Subject: Computer Applications
Practical   **2 Hours/Week**   **1 credit**
Relational Database Management Systems Lab <u>Course</u>
<u>**Outcome:**</u>

1) To explain basic database concepts, applications, data models, schemas and instances.
2) To demonstrate the use of constraints and relational algebra operations. IV. Describe the basics of SQL and construct queries using SQL.
3) To emphasize the importance of normalization in databases.
4) To facilitate students in Database design .
5) To familiarize issues of concurrency control and transaction management.

<u>**Practical Question Bank**</u>

Consider the relational schema for part of the DreamHome case study is:
Branch (branchNo, street, city, postcode)
Staff (staffNo, fName, IName, position, sex, DOB, salary, branchNo)
PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo, staffNo, branchNo)
Client (clientNo, fName, IName, telNo, prefType, maxRent, eMail)
PrivateOwner (ownerNo, fName, IName, address, telNo, eMail, password)
Viewing (clientNo, propertyNo, viewDate, comment)
Registration (clientNo, branchNo, staffNo, dateJoined)

1. Create a database with name "DreamHome" and now create all the tables listed above with constraints.
2. Insert a new row into the table supplying data for all columns.
3. Modify data in the database using UPDATE
4. Delete data from the database using DELETE
5. Changing a table definition using ALTER
6. Removing a table using DROP
7. Removing rows in table using TRUNCATE
8. Create an index and removing an index
9. Practice other standard SQL commands for creating, modifying, displaying data of tables.
10. List full details of all staff.
11. List all staff with a salary greater than £10000.
12. List the property numbers of all properties that have been viewed.
13. Produce a list of salaries for all staff, showing only the staffNo, fName, IName, and salary details.
14. List all cities where there is either a branch office or a property for rent.
15. List all cities where there is a branch office but no properties for rent.

16. List all cities where there is both a branch office and at least one property for rent.
17. List the names and comments of all clients who have viewed a property for rent.
18. Produce a status report on property viewings.
19. List complete details of all staff who work at the branch in Glasgow.
20. List the addresses of all branch offices in London or Glasgow.
21. List all staff with a salary between £20,000 and £30,000.
22. Identify all clients who have viewed all properties with three rooms.
23. How many properties cost more than £350 per month to rent?
24. How many different properties were viewed in May 2013?
25. Find the total number of Managers and the sum of their salaries.
26. Find the minimum, maximum, and average staff salary.
27. Find the number of staff working in each branch and the sum of their salaries.
28. List all managers and supervisors.
29. Find all owners with the string 'Glasgow' in their address.
30. List the details of all viewings on property PG4 where a comment has not been supplied.
-

Department Of Computer Science &Applications Trans
Disciplinary Elective
Semester-IV
Subject: <mark>Web Designing</mark>
Paper- II (B)

COURSE
CODE:CA402

No. of Lectures:
15 No. of
Credits: 1

Course Outcome:

1. Understand the principles of creating an effective web page, including an in-depth consideration of information architecture.
2. Become familiar with graphic design principles that relate to web design and learn how to implement theories into practice.
3. Develop skills in analyzing the usability of a web site.
4. Understand how to plan and conduct user research related to web usability.
5. Learn the language of the web: HTML and CSS.
6. Learn CSS grid layout and flexbox.
7. Learn techniques of responsive web design, including media queries.

Syllabus

UNIT-I
<mark>Working with Links and Lists:</mark>
Creating Hyperlinks-URL (Uniform Resource Locator)-Creating Image Maps-Creating List-Ordered list-Unordered List-Definition List.

UNIT-II
<mark>Working with Images:</mark>
Images in web pages-graphic formats-graphic color-Creating Images-Adding an Image to web page-Allocating space for Image-Adding Borders to Images-Aligning the Text and Images-Setting page background Images.

UNIT-III
<mark>Working with Frames & Forms:</mark>
Creating Frames- Vertical Frames- Horizontal Frames. Creating HTML Forms with Example.

B.A III
YEAR
Semester-V
Subject: Computer Applications
Paper- III (A) JAVA PROGRAMMING-I

P.P.W: 6(4Th+2Pr)

COURSE CODE:CA501   **Credits: 3Th+1Pr**

Course Outcome:

1. Each course outcome is followed in parentheses by the Program Outcome to which it relates. - Write, compile, and execute Java programs that may include basic data types and control flow constructs using J2SE or other Integrated Development Environments (IDEs) such as Eclipse, NetBeans, and JDeveloper.
2. Write, compile and execute Java programs using object oriented class structures with parameters, constructors, and utility and calculations methods, including inheritance, test classes and exception handling.
3. Write, compile, and execute Java programs using arrays and recursion.
4. Write, compile, and execute Java programs manipulating Strings and text documents.
5. Write, compile, execute Java programs that include GUIs and event driven programming. Write a final project that may be selected from among the following: applets for inclusion in web pages; applets to access enterprise data bases in robust, enterprise three level applications; secure communications over the internet; or an approved project chosen by the student.

Syllabus

UNIT-1: Introduction to Java, language and Importance of Java on Internet. Java Buzz words, object oriented programming and Java Demo on sample program and Compiling the program, Data Types.

UNIT-2 Variables [Declaration Dynamic Initialization and scope & lifetime] Type conversion, Arrays, operators [Arithmetic, Relational Boolean] Java control statement Iteration statements, Iteration statements Jump statement Introduction to Classes and objects.

UNIT-3 Introduction to methods and Constructors Garbage Collection Introduction to Overloading. Overloading in detail, Introduction to Inheritance, Inheritance in detail. Introduction to Multiple hierarchy and in detail, Introduction to Abstract classes in detail.

UNIT-4 Packages and Inter faces :- Introduction to package and in detail, Access protection with examples, Important packages Interfaces: Defining and Implementing and with examples,

Exception Handling, Fundamentals, different types of Exception handling use of Try, catch, throw, Throws and Finally.

Suggested Books:
Prescribed Books:
 Programming with
Java:E.Balagurusamy Java Complete
reference.

Reference Books:
Java 2.0:Ivan Bayross
Java Tutorial:Sun microsystems
Special edition using Java 2:Joseph L.Weber

B.A III Year Practical Question Bank
Subject: Computer Applications
Paper: III (A) JAVA PROGRAMMING-1

Course Outcome:

1. Implement Object oriented features using Java
2. Apply the concept of polymorphism and inheritance.
3. Implement exception handling
4. Develop network and window application using awt and swings.

Practical Question Bank

1. Write a Java program using various control statements.
2. Write a Java program using Constructors.
3. Write a Java program to demonstrate Multi-level Inheritance.
4. Write a Java program for Matrix Multiplication.
5. Using Packages in Java, Write a appropriate…
6. Write a Java code for implementation of Interfaces.
7. Explain "Exception handling" with suitable example.
8. Implementing String handling functions using Java program.
9. Write a java program to determine the sum of the following harmonic series for a given value of 'n'.
   $1+ \frac{1}{2} + \frac{1}{3} + \qquad + 1/n.$
10. Write a program to perform the following operations on string through interactive input.
    (a) Sort given Strings in alphabetical order.
    (b) Check whether one String is Sub String of another String or not.

B. A III
YEAR
Semester-V
Subject: Computer Applications
Paper- IV (A) –WEBTECHNOLOGIES-I

P.P.W: 5(3Th+2Pr)
Credits: 2Th+1Pr

COURSE CODE:CA501

Course Outcome:

1. Gain knowledge of client side scripting, validation of forms and AJAX programming.
2. Have understanding of server side scripting with PHP language.
3. Have understanding of what is XML and how to parse and use XML Data with Java.
4. Create applications by using the concepts like JSP and Servlet

Syllabus

UNIT I : Introduction to Internet basics, Clint& Server technology Inter connectivity. Web Client/ Browser, available. Introduction to HTML (Hypertext markup language), HTML commands, Titles & Footers, Text formatting, Text Styles
UNIT II : , Classification of Tags-Padded tags Unpadded tags-Formatting Tags-Hyper Links-Heading tags.
 Lists, Text effect Adding Graphics to Html document, Tables, Linking of documents, Frames some exercises. Forms-Meaning of Forms-Creation of Forms.

UNIT-III : Dynamic HTML Programming Introduction-Difference between

HTML and DHTML.Static and Dynamic tags, Procedural and Non-Procedural Programming in DHTML, Event Handling**.**

UNIT IV : Cascading Style Sheets (CSS)-Document Object Model (DOM)-Change like Style, Text, Graphics, Placement, Creating multimedia effects with Filters-Different types of Errors, Runtime Errors, System Errors etc.

Suggested Books:
Prescribed Books: HTML, DHTML, JAVASCRIPT, PERL, CGI:Ivon bayross The
 complete reference Webdesign:Thomas A.Powel.

Reference Books:
Scripting Language and Web designing:R.singh,Mamatha varma.s.Mahindru
World wide Web :Rick Stout

B.A III Year Practical Question Bank
Subject: Computer Applications
**Paper – WEBTECHNOLOGIES-I**

Course Outcome:

1. Students are able to develop a dynamic webpage by the use of java script and DHTML.
2. Students will be able to write a well formed / valid XML document.
3. Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
4. Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
5. Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database.

Practical Question Bank

1. 1.Create a web page illustrating text formatting tags.
2. Create a web page to demonstrate font variations.
3. Prepare a sample code to illustrate three types of lists in HTML.
4. Using tables create your Curriculum Vitae.
5. Using Table related tags align the images

6. Explain hyper link example with target attribute as follows.
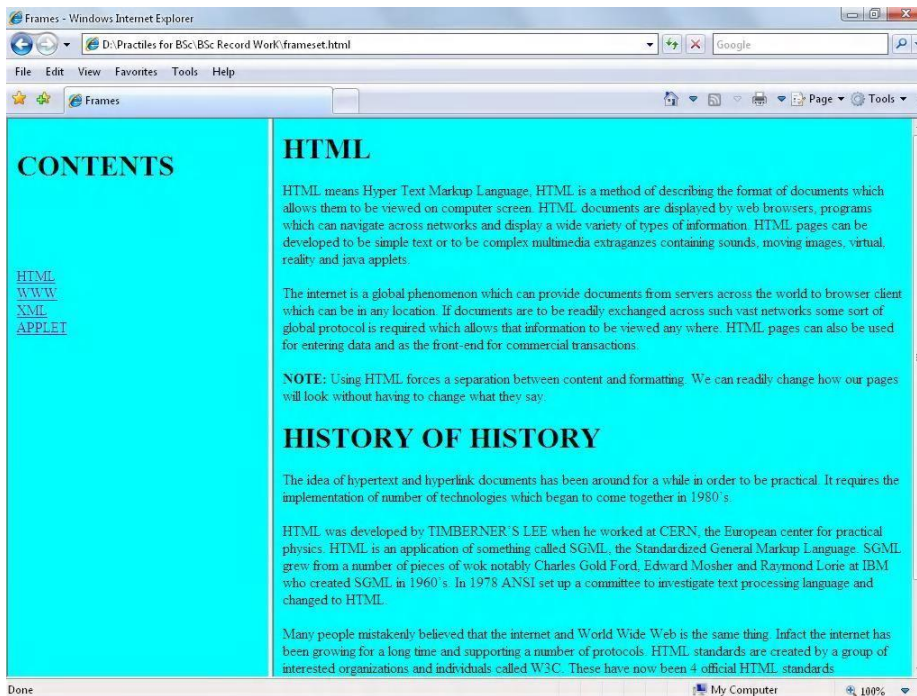
7. Illustrate with example the horizontal rulers in your page.

8. Create a Web page to display the timetable of your class with the following format and use all the attributes associated with table tag.

9. Create a Web page to display the following table using all the attributes of table tag:

IMG LOGO		POPULATION			
		STATE 1		STATE 2	
		M	F	M	F
2005	Lit				
	illit				
2006	Lit				
	illit				

9. Create a web page of Employee information form, when the information is submitted, message should be displayed.

10. a) Create a web Page to display 5 images using all the attributes of the image tag

11. b) Create a Web Page to display ordered list, unorder list, and definition list.

12. Create a Web Page to display frame with the following format, Use necessary attributes for the tags. When the user clicks the link in the left frame the contents should be displayed in the right frame.

HEADING	
Order list Unorder list Definition list	Display the clicked page here

13. Create a Web page to display Overlapping of Images. The images should scroll block.

14. Create a web Page to display overlapping text .
15. Create a Web page with an image, when the mouse is doubled clicked new image should replace the existing.

16. Create a Web Page to display text aligned with images.
17. Create a Web page with different text decoration attributes.
18. Create a Web page with external style sheet.
19. Create a Web page to display students registration form with the following format:

STUDENTS APPLICATION FORM

Student Name:

Father's Name:

Address:

Course: B.Sc. B.Com, Comp.   Combination   Reg. Vocation

Gender: O Male   O Female
Qualification:

Sports interested in

☐ Tennis   ☐ Cricket   ☐ Football   ☐ Long Jump

Reset

Submit

Scrolling message

Government College for Women Begumpet, Hyderabad-500016 (An Autonomous college of Osmania University)
Re-Accredited by NAAC with 'B' Grade

B.A III
YEAR
Semester-VI
Subject: Computer Applications
Paper- III (B) <mark>JAVA PROGRAMMING</mark>-II

P.P.W: 6(4Th+2Pr)
COURSE CODE:CA601   **Credits: 3Th+1Pr**

<u>**Course Outcome:**</u>

1. Each course outcome is followed in parentheses by the Program Outcome to which it relates. - Write, compile, and execute Java programs that may include basic data types and control flow constructs using J2SE or other Integrated Development Environments (IDEs) such as Eclipse, NetBeans, and JDeveloper.
2. Write, compile and execute Java programs using object oriented class structures with parameters, constructors, and utility and calculations methods, including inheritance, test classes and exception handling.
3. Write, compile, and execute Java programs using arrays and recursion.
4. Write, compile, and execute Java programs manipulating Strings and text documents.
5. Write, compile, execute Java programs that include GUIs and event driven programming. Write a final project that may be selected from among the following: applets for inclusion in web pages; applets to access enterprise data bases in robust, enterprise three level applications; secure communications over the internet; or an approved project chosen by the student.

<u>**Syllabus**</u>

UNIT-1 : <mark>Built-in exception</mark>, I/P-O/P: I/O Basic reading console I/P & writing Console O/P Reading and writing String Handling ,String constructor, String operators, Character Extraction String comparison searching and Modifying Strings, Data conversion, String buffer, Applet basics ,Architecture of Applets

<mark>**UNIT-2 :** Some simple Applet display methods ,requesting Repainting,</mark> using the Status window ,understanding the HTML Applet tags. Passing parameter to Applet event model, Event model, Event classes, Sources of Events, Event listener Interfaces, Adapter classes and Inner classes.

UNIT-3 : <mark>Introduction to AWT classes window fundamentals,</mark> Frame windows, creating a Frame window, Creating Windowed program and Sizing graphics, Color and use of Font Methics using AWT controls, Lists and using Buttons, Check box group and using Buttons, Check box group and Choice control using List and managing Scroll bars, using Text file and Text area.

UNIT-4: <mark>Understanding Layout Management: Form layout, Border layout</mark>, Menus and Dialog boxes, Files dialog and Exploring the Controls, Menus and Layout Manager.

<u>Suggested Books:</u>
<u>Prescribed Books:</u>
 Programming with Java:
E.Balagurusamy Java Complete
reference:

<u>Reference Books:</u>
Java 2.0: Ivan Bayross
Java Tutorial: Sun Microsystems
Special edition using Java 2: Joseph L.Weber

SEMESTER-V
B.A III Year Practical Question Bank
Subject: Computer Applications
Paper: III (A) <mark>JAVA PROGRAMMING</mark>-II

Course Outcome:

1. Implement Object oriented features using Java
2. Apply the concept of polymorphism and inheritance.
3. Implement exception handling
4. Develop network and window application using awt and swings.

Practical Question Bank

1. Demonstrate Applets in Java.
2. Write a suitable code in Java for event handling.
3. Write a code in Java using AWT Controls.
4. Write a Java code using Super Keyword.
5. Explain the abstract classes with suitable examples.
6. Explain function overloading with an example.
7. Menu-Driven program
8. Write an Applet program to design a simple calculator.
9. Write a program to read a text and count all the occurrences of a given word. Also display their positions.
10. Write an applet illustrating sequence of events in an Applet.
11. Illustrate the method of overriding in Java.

Government College for Women Begumpet, Hyderabad-500016 (An Autonomous college of Osmania University)
Re-Accredited by NAAC with 'B' Grade

B. A III
YEAR
Semester-VI
Subject: Computer Applications
Paper- IV (B) –WEBTECHNOLOGIES-II

COURSE CODE:CA602

P.P.W: 5(4Th+2Pr)
Credits: 2Th+1Pr

Course Outcome:

1. Gain knowledge of client side scripting, validation of forms and AJAX programming.
2. Have understanding of server side scripting with PHP language.
3. Have understanding of what is XML and how to parse and use XML Data with Java.
4. Create applications by using the concepts like JSP and Servlet

Syllabus

UNIT I: Introduction to Java Script Java Script in Web pages, Advantages of Java Script, writing Java Script into HTML, Building Java Script syntax, Operators & Expressions in Java Script . Conditional checking statements, Loops & Functions in Java Script Dialog boxes

UNIT II:
Variables, Arrays, Objects in Java Script: Data and objects in java script, Regular expressions, Exception Handling, Built in objects, Events. Dynamic HTML with Java Script: Data validation, Opening a new window, Messages and
Confirmations, The status bar, writing to a different Frame, Rollover buttons, Moving images.

UNIT III:
Data for Web applications: Basic XML, Document type definition,
XML Schema, Active Server Pages and Java: Active Server Pages,
Java.
XML: Defining Document Object Model, Presenting XML.

UNIT IV:
Protocols: Protocols, IP and TCP, Hyper Text Transfer Protocol, Common Gateway Interface, The Document Object Model, introducing the Document Object Model, Exercises

Suggested Books:
Prescribed Books:
HTML, DHTML, JAVASCRIPT, PERL, CGI: Ivon bayross

The complete reference Webdesign: Thomas A. Powel

Reference Books:
Scripting Language and Webdesigning: R.singh, Mamatha varma.s.Mahindru

Government College for Women Begumpet, Hyderabad-500016 (An Autonomous college of Osmania University)
Re-Accredited by NAAC with 'B' Grade
SEMESTER-VI
B.A III Year Practical Question Bank
Subject: Computer Applications
Paper – WEBTECHNOLOGIES-II

Course Outcome:

1. Students are able to develop a dynamic webpage by the use of java script and DHTML.
2. Students will be able to write a well formed / valid XML document.
3. Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
4. Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
5. Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database.

Practical Question Bank

1. Create a java script program to accept the first, middle, last names of user

   and print them.

   Evaluate the following:

   a. "10 "+ " 20"

   b. (10<8)?10:8

   c. J=(i++)+(--i)+(++i)+(i++) where i=2

2. Write a java script program to add two numbers.

3. Write a java script program to find the factorial of given number.

4. Write a java Script program to print all prime numbers.

5. Write a java script program to sort the array (Bubble Sort).

6. Write a java script program to evaluate the following mathematical

   Expression 1+2/2!+3/3!+   +n/n!.

7. Create an Online Bio-Data Form for the Current Employees in the organization.

8. Design the simple Calculator.

9. Write a java script program to "Wish a user " at different hours of a day.

10. Prompt a user for the cost price and selling price of an article and output the profit

and loss percentage.

11. Create a web page of customer profile for data entry of customer's in a Hotel, The profile should include Name, Address, Age, gender, Room Type (A/C, Non-A/C or Deluxe), Type of payment (Cash, Credit/Debit Card or Coupons).

12. Create an Online Bio-Data Form for the Current Employees in the organization.

13. Write DHTML program to give different colors for different heading tags.

14. Using CSS invert the behavior of the <h1> to <h6> tags.
15. Create a sample code to illustrate the Inline style sheet for your web page.
16. Create a sample code to illustrate the External style sheet for your web page.
17. Create a sample code to illustrate the embedded style sheet for your web page.
18. Create a sample code to illustrate the procedure of creating user defined classes in CSS.

B.A (C/A) I, II & III Year Examination
Subject: Computer Applications
MODEL PAPER

Time: 2 1/2hrs   Max Marks: 75

SECTION-A

I Answer all the questions.   4X10=40

    1. a) Question from Unit-I.
          (OR)
      b) Question from Unit-I.

    2. a) Question from Unit-II.
          (OR)
b)   Question from Unit-II.
3.   a) Question from
Unit-III.
          (OR)
b)   Question from Unit-III.
4.   a) Question from Unit-IV.
          (OR)
      b) Question from Unit-IV.

SECTION-B

II. Answer Any 5 questions.   5X5=25
1. 1) Question from Unit-I.
2. 2) Question from Unit-I.
3. 3) Question from Unit-II.
4. 4) Question from Unit-II.
5. 5) Question from Unit-III.
6. 6) Question from Unit-III.
7. 7) Question from Unit-IV.
8. 8) Question from Unit-IV.

SECTION-C

III Answer Any 5 questions.   5X2=10
1) Question from Unit-I.
2) Question from Unit-I.
3) Question from Unit-II.
4) Question from Unit-II.
5) Question from Unit-III.
6) Question from Unit-III.
7) Question from Unit-IV.

8) Question from Unit-IV.

<div align="center">

Trans Disciplinary Elective
PATTERN OF EXAMINATION

</div>

Question Paper Pattern for Assignments and Theory Examinations of Semester III &IV for all combinations.

Assignments

1. 3 Assignments of 10 marks each and best 2 out of 3 will be considered.

Semester Examination

1. To be held in the month of October and March/April months.
2. The Time Duration of Semester examination is 2 hrs.
3. 30 marks are allotted for the main exam for each semester
4.   Section – A: 12 Multiple Choice Questions each carries 1 mark.
        12x1=12M B: 6 out of 9 questions each question carries 3 marks
                                    6x3=18M

        Total marks=30 (End Semester Exam) + 20(Assignments)

<u>Practical Examinations</u>

1. Practical examinations will be held at the end of each Semester.
2. 50 marks are allotted for the Practical examination consisting of External and Internal Evaluation.
3. Practical Question Bank is prepared & provided to the students from which practical will be conducted.
4. Practical shall be conducted in each Semester as per the Syllabus and Time table.