



बस्तर विश्वविद्यालय, जगदलपुर (छ.ग.)  
BASTAR VISHWAVIDYALAYA, JAGDALPUR (C.G.)

# SYLLABUS B.C.A. PART-II SESSION 2020-21

बस्तर विश्वविद्यालय, जगदलपुर (छ.ग.)

धरमपुरा, जगदलपुर, जिला-बस्तर (छ.ग.) 494001

वेबसाईट [www.bvvdjp.ac.in](http://www.bvvdjp.ac.in) दूरभाष-07782-229037 फ़ैक्स 229297

## SCHEME OF EXAMINATION 2019-2020

BCA PART- II

Subject Code	Subject Paper	Theory Marks		Internal Marks		Teaching Load per Week		
		Max. (A)	Min. (B)	Max. (C)	Min. (D)	L	T	P
BCA201	Calculus and Differential Equations	80	27	20	8	4	2	-
BCA202	Database Management System	80	27	20	8	4	2	-
BCA203	Programming in 'C++'	80	27	20	8	4	2	-
BCA204	Computer Networks	80	27	20	8	4	2	-
BCA205	Operating Systems with Linux	80	27	20	8	4	2	-
BCA206	Foundation Course	80	27	20	8	4	2	-
BCA207	LAB IV: Programming Lab in 'C++'	100	50	40	16	-	-	3x2
BCA208	LAB V: Database Management System Lab	100	50	40	16	-	-	2x2
BCA209	LAB VI: Operating System Lab	100	50	20	8	-	-	1x2
<b>TOTAL</b>		<b>780</b>	<b>312</b>	<b>220</b>	<b>88</b>			
<b>GRAND TOTAL</b>	<b>(PAPER + INTERNAL)</b>	<b>(A+C)</b> <b>1000</b>		<b>(B+D)</b> <b>400</b>				

- Student will have to pass individually in all theory, practical and sessional.

K. Anas  
27/4/19

Pamyan  
27/4/19

~~4007~~  
27-04-19

P. Jate  
27-4-19

Bah  
27/4/19

Sumar  
27-04-19

**Calculus and Differential Equations**  
**Subject Code - BCA-201**

**Max Marks : 80**

**Min Marks : 27**

**Note :** The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice. Only Simple calculators allowed not scientific calculator.

**Differentiation**

**UNIT- I**

Limits -Definition of limits, Continuity of one variable, Types of continuity, Properties of continuous function: Borel's Theorem, Boundedness Theorem, Mostest Theorem, Intermediate value theorem, Differentiability of function(s) of one variable

**UNIT - II**

Differentiation of Functions, Differentiation of functions of functions, parametric functions, product of functions, function in Product and quotient form, Logarithmic differentiation, Differentiation of Parametric functions. Higher order derivative, Maxima and Minima

**Integration**

**UNIT - III**

Indefinite Integral- Basic integration Formulas, Trigonometric Integrals, Integration by Parts, Integration by substitution

**UNIT - IV**

Definite Integrals- Introduction, Properties of definite integrals, Problem based on properties of definite integrals

**Differential Equation**

**UNIT -V**

Introduction to differential equation: Definition, order and degree of differential equation, derivation of a differential equation, general and particular solution of differential equation, separation of variables.

**TEXTBOOK:**

1. Calculus and Statistical Analysis: H.K. Pathak
2. Calculus : B.R. Thakur
3. Differential Equation: H.K.Pathak

**REFERENCE:**

1. Differential Calculus : Gorakh Prasad
2. Differentiation & Integration : H.K Pathak
3. Integral Calculus: Gorakh Prasad
4. Differential Equation : Gorakh Prasad
5. Calculus: Rey & Sharma

*Kiran*  
27/4/19

*Bhuvan*  
27/4/19

*Prasanna*  
27-34-19

*Prasanna*

*Prasanna*  
27/4/19

*Prasanna*  
27-04-2019

*Prasanna*  
27.4.19

**Database Management System**  
**Subject Code - BCA-202**

**Max Marks : 80**

**Min Marks : 27**

**Note :** The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice. Only Simple calculators allowed not scientific calculator.

**UNIT - I: Overview of Database Management**

Data, Information and knowledge, Increasing use of data as a corporate resource, data processing verses data management, file oriented approach verses database oriented approach to data management; data independence, database administration roles, DBMS architecture, different kinds of DBMS users, importance of data dictionary, contents of data dictionary, types of database languages. Data models: network, hierarchical, relational.

**UNIT - II: Relational Model & Relational Algebra**

Entity-Relationship model as a tool for conceptual design-entities, attributes and relationships. ER diagrams; Concept of keys; Case studies of ER modeling Generalization; specialization and aggregation. Converting an ER model into relational schema. Extended ER features, Introduction to UML, Representation in UML diagram (Class Diagram etc.).

**UNIT - III: Relational Database Design**

**Relational Algebra:** select, project, cross product different types of joins (inner join, outer joins, self-join); set operations, Tuple relational calculus, Domain relational calculus, Simple and complex queries using relational algebra, stand alone and embedded query languages.

**UNIT - IV: Structured Query Language (SQL)**

Normalization concept in logical model; Pitfalls in database design, update anomalies: Functional dependencies, Join dependencies, Normal forms (1NF, 2NF, 3NF). Boyce Codd Normal form, Decomposition, Multi-Valued Dependencies, 4NF, 5NF. De-normalization.

**UNIT - V: Query Processing and Security**

Introduction to SQL constructs (SELECT...FROM, WHERE... GROUP BY... HAVING... ORDERBY...), INSERT, DELETE, UPDATE, DROP, VIEW definition and use, Temporary tables, Nested queries, and correlated nested queries, Integrity constraints: Not null, unique, check, primary key, foreign key, references, Inner and Outer Joins. **Query Processing:** Parsing, translation, optimization, evaluation and overview of Query Processing. **Protecting the Data Base:** Integrity, Security and Recovery. Domain Constraints, Referential Integrity, Assertion, Triggers, Security & Authorization in SQL.

**BOOKS RECOMMENDED:**

1. **Database System Concept:** A. Silberschatz, H.F. Korth and S. Sudarshan, TMH
2. **Fundamentals of Database Systems:** Elmasri & Nawathe, Pearson Education
3. **An Introduction to Database Systems:** C. J. Date, AWL Publishing Company
4. **SQL, PL/SQL:** Ivan Bayross, BPB Publication
5. **An Introduction to database systems:** Bipin Desai, Galgotia Publication.
6. **Database Management System:** A. K. Majumdar & P. Bhattacharya, TMH

*Shris*  
27/4/19

*Benjamin*  
27/4/19

*Date*  
27.4.19

*[Signature]*  
27.4.19

*[Signature]*

*Rahul*  
27/4/19

*[Signature]*  
27.4.19

**Programming in "C++"**  
**Subject Code - BCA-203**

**Max Marks : 80**

**Min Marks : 27**

**Note :** The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice. Only Simple calculators allowed not scientific calculator.

**UNIT - I: Language Fundamental**

**Overview of OOP:** The Object Oriented paradigm, Basic concepts of OOP, Benefits of OOP, Object oriented languages, Application of OOP

**Overview of C++:** History of C++, **Data Types:** Built-in data types, User-defined data types, Derived data types. **Constants and Variables:** symbolic constants, Dynamic initialization of variable, Reference variable. **Operators in C++.** **Control Structures:** if-else, nested if-else, while, do-while, for, break, continue, switch, goto statement.

**UNIT - II: Structure & Function**

**Structures:** A Simple structure, defining a structure variable, Accessing structure's member, Enumeration data type.

**Function:** Function Declaration, Calling Function, Function Definition, **Passing Arguments to function:** Passing Constant, Passing Value, Reference Argument, Structure as argument, Default Argument.

**Returning values from function:** return statement, Returning structure variable, Return by reference. Overloaded Function, Inline Function.

**UNIT - III: Object Classes and Inheritance**

Object and Class, Defining the class and its member, Making an outside function inline, nesting of member function, array as class member, structure and classes.

**Memory allocation:** memory allocation for objects, new and delete operator, static data member, static member functions, object as function argument.

**Constructor & Destructor:** Null and default constructor. Parameterized constructor, Constructor with default argument, copy constructor, class destructors,

**UNIT - IV: Pointers and Inheritance**

**Pointers:** Introduction, & and \* operator, pointer to object, this pointer, pointer to derived class.

**Inheritance:** Introduction to inheritance, Types of inheritance, function overriding, Constructor in Derived class. Access specifiers: public, private, protected.

**UNIT - V: Polymorphism**

**Dynamic polymorphism:** Virtual function, Pure Virtual Function, Abstract class.

**Static Polymorphism:** Operator keyword, overloading unary operators (++ (pre increment and post increment), --) using operator function, overloading binary operators (+, -, ==, >=, <=, +=, <, >, []), Friend function, Friend class, overloading binary operators using friend function.

**RECOMMENDED BOOKS :**

1. **Object Oriented Programming with C++ :** E. Balagurusamy, The McGraw-Hill
2. **Let Us C++:** Yesvant Kanetkar, BPB Publications
3. **The C++ Programming Language:** Bjarne Stroustrup, Addison Wesley.
4. **Object Oriented Programming in C++ :** Robert Lafore, Galgotia Publications.

*Anon*  
27/4/19

*Panjan*  
27/4/19

*[Signature]*  
27/4/19

*[Signature]*

*Bah*  
27/4/19

*[Signature]*  
27/4/19

*[Signature]*  
27/4/19

**Computer Networks**  
**Subject Code - BCA-204**

**Max Marks : 80**

**Min Marks : 27**

**Note :** The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice. Only Simple calculators allowed not scientific calculator.

**UNIT - I -Introduction to Computer Networking**

Data Communication, Networks - Distributed Processing, Network Criteria, Applications; Protocols and Standards, Standard Organization, Line Configuration - Point to Point, Multi Point; Topology - Mesh, Star, Tree, Bus, Ring, Hybrid; Transmission mode, Categories of Network - LAN, MAN, WAN, Inter Networks.

**UNIT - II -Transmission of Digital Data**

Analog and Digital, digital data transmission - parallel transmission, serial transmission, DTE-DCE interface - data terminal equipment, data circuit terminating equipment, standards, modems- Transmission rate, Modem standards.

**UNIT - III- The OSI Model**

ISO organization, The model - Layered architecture, functions of the layers -Physical layer, Data Link layer, Network layer, Transport layer, session layer, Presentation layer, Application layer

**UNIT - IV TCP/IP Model & Protocols**

The TCP/IP reference model, comparison of TCP/IP & OSI, Introduction to Internet - ARPANET, Architecture of Internet, Client server model, www, IP Address Classes, Protocols: IP, HTTP, TCP, FTP, ARP.

**UNIT - V Network Security**

Introduction of Network Security and it's importance. **Cryptography:** Definitions, **Symmetric Key Cryptography:** Traditional Ciphers, Simple modern Ciphers, **Asymmetric Key Cryptography:** RSA, Security Services, Digital Signatures.

**BOOKS RECOMMENDED:**

1. Introduction to Data communication & Networking - Behrouz & Forouzan
2. Computer Networking - Andres & Tanenbaum

*Amr*

*Beaujeu*  
27/4/19

*Amr*  
27-4-19

*Amr*

*Beau*  
27/4/19

*Amr*  
27-04-2019

*Amr*  
27-4-19

**Operating Systems with Linux**  
**Subject Code - BCA-205**

**Max. Marks: 80**

**Min Marks : 27**

**Note :** The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice. Only Simple calculators allowed not scientific calculator.

**UNIT - I: Introduction**

Defining operating system, History and Evolution of operating system, **Basic Concepts:** batch processing, spooling, multiprogramming, multiprocessor system, time sharing, real time systems, Functions and Goals of operating system.

**UNIT - II: Process Management**

Process concept, Process Control Block, **Process State:** State Transition Diagram, **Scheduling Queues:** Queuing Diagram, Types of schedulers-context switching and dispatcher, various types of CPU scheduling algorithms and their evaluation, multilevel queues and multilevel feedback queues.

**UNIT - III: Memory Management**

Preliminaries of memory management, Contiguous memory allocation, fragmentation, partition allocation policies, compaction, Non-Contiguous memory allocation, Paging, Segmentation, **Virtual Memory:** Demand paging, Swapping, **Page replacement policies:** FIFO, Optimal, LRU, MRU.

**UNIT - IV: Introduction to UNIX**

Introduction to Multi-user System, Emergency and history of Unix, Feature and benefits, Versions of Unix. **System Structure:-**Hardware requirements, Kernel and its function, introduction to System calls and Shell.

**File System :** Feature of Unix File System, Concept of i-node table, links, commonly used commands like who, pwd, cd, mkdir, rm, ls, mv, lp, chmod, cp, grep, sed, awk, pr, lex, yacc, make, etc. Getting started (login / logout). **Vi Editor:-**Intro to text processing, command and edit mode, invoking vi, command structure, deleting and inserting line, deleting and replacing character, searching strings.

**UNIT - V: Shell Programming**

Introduction to shell feature, wild card characters, i/out redirections, standard error redirection, system and user created shell variables, profile files, pipes/tee, background processing, command line arguments, command substitution, read statement, conditional execution of commands, special shell variables \$ #, #?, \$\* etc. Shift commands, loops and decision making- for, while and until, choice making using case...esac, decision making if ....fi, using test, string comparison, numerical comparison, logical operation, using expr.

**BOOKS RECOMMENDED:**

1. Operating System Concepts, Abraham Silberschatz, Peter B. Galvin and Greg Gagne (Wiley India Edition)
2. Modern Operating System, Andrew .S. Tanenbaum, (PHI)
3. UNIX Complete Reference

*Kris*  
27/4/19

*Ramya*  
28/4/19

*Teja*  
27-4-19  
*Dee*  
27-4-19

*Blah*  
27/4/19  
*Shree*  
27-4-19

**Foundation Course**  
**Subject Code - BCA - 206**

**Max. Marks: 80**

**Min Marks : 27**

**Note :** The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice. Only Simple calculators allowed not scientific calculator.

**OBJECTIVE**

This course is designed to make the students acquainted with Indian History and Culture. To make students aware of their fundamental rights & duties and to have the knowledge of parliamentary form of Government. To groom students and develop their professional skills.

**Unit I**

Indian Art, meaning of art, features of indian art, elementary knowledge of paintings, music, dancing, sculpture archeology, iconography & other social arts.

**Unit II**

Indian Literature, Ancient Indian Literature, Elementary knowledge of Vedic Literature, Mahabharata, Ramayan and other main granthas.

**Unit III**

Indian Freedom Struggle : Freedom Struggle of 1857, National Consciousness, non-cooperation movements. Civil disobedient movement quit India movement, contribution of revolutionaries in freedom struggle.

**Unit IV**

Indian Constitution : Introduction, main features of constitution, fundamental rights.  
Parliamentary Government: Meaning, Features, Rajya Sabha, Lok Sabha.

**Unit V**

Communication: Process, Channels, Barriers.  
Listening: Types, Purpose, Barriers, Effective Listening Strategies.  
Job Interviews: Résumé Writing, Group Discussion, Job Application Writing, Interview Preparation.

**BOOKS RECOMENDED:**

- Indian Culture the book sponsored by M.P. Hindi granth Academy .
- Parliamentary Procedure in India by A.R.Mukherjea
- Effective Technical Communication by M Ashraf Rizvi

*K. Anand*  
27/4/19

*Rangini*  
27/4/19

*Dr. D. D. D.*  
27-4-19

*Bah*  
27/4/19

*Shree*  
27/4/19



**PRACTICAL WORK BCA II**  
**BCA-207 - LAB IV: Programming Lab in 'C++'**

**1 Scheme of Examination:-**

Practical examination will be of 3 hours duration. The distribution of practical marks will be as follows

Program 1	-	20
Program 2	-	20
Program 3	-	20
Viva	-	25
[ Practical Copy + Internal Record ]	-	15
Total	-	100

- In every program there should be comment for each coded line or block of code
- Practical file should contain printed programs with name of author, date, path of program, unit no. and printed output.
- All the following programs or a similar type of programs should be prepared

**List of Practical**

**LOOPS, DECISIONS, NESTED METHOD, MEMBER FUNCTION DEFINED OUTSIDE CLASS BODY:**

**1. Write program to generate following pattern**

a) A B C D E F G  
A B C E F G  
A B F G  
A G

c) \*  
\* \*  
\* \* \*

b) 1  
1 2  
1 2 3  
1 2 3 4

d) 1  
1 2 1  
1 3 3 1  
1 4 6 4 1

- Write member functions which when called asks pattern type; if user enters 11 then a member function is called which generates first pattern using for loop. If user enters 12 then a member function is called which generates first pattern using while loop. If user enters 13 then a member function is called which generates first pattern using do-while loop. If user enters 21 then a member function is called which generates second pattern using for loop and so on.
- Write program to display number 1 to 10 in octal, decimal and hexa-decimal system.
- Write program to display number from one number system to another number system. The program must ask for the number system in which you will input integer value then the program must ask the number system in which you will want, output of the input number after that you have to input the number in specified number system and program will give the output according to number system for output you mentioned earlier.

**Array**

- Write a program using function to add, subtract and multiply two matrices of order 3x3. You have to create one function for addition, which accepts three array arguments. First two array arguments are matrices to add and third matrix is destination where the resultant of addition of first two matrixes is stored. In similar way create functions for matrix subtraction and multiplication.
- Create a single program to perform following tasks without using library functions:
  - To reverse the string accepted as argument.
  - To count the number of characters in string passed as argument in form of character array.
  - To copy the one string to other string; passed as arguments in form of source character array and destination character array without using library function.
  - To count no. of vowels, consonants in each word of a sentence passed as argument in form of character array.

*Prateek*  
27/4/19

*Pratyaksh*  
27/4/19

*Pratyaksh*  
27-04-19

*Pratyaksh*

*Pratyaksh*  
27/4/19

*Pratyaksh*  
27-04-19

### Class, Object, Array of object, Object Using Array

7. Create a class Student having data members to store roll number, name of student, name of three subjects, max marks, min marks, obtained marks. Declare an object of class student. Provide facilities to input data in data members and display result of student.
8. Create a class Student having data members to store roll number, name of student, name of three subjects, max marks, min marks, obtained marks. Declare array of object to hold data of 3 students. Provide facilities to display result of all students. Provide also facility to display result of specific student whose roll number is given.
9. Create a class Sarray having an array of integers having 5 elements as data member provide following facilities:
  - a) Constructor to get number in array elements. b) Sort the elements. c) Find largest element
  - b) Search for presence of particular value in array element.

### Static member function

10. Create a class Simple with static member functions for following tasks:
  - a) To find factorial by recursive member function.
  - b) To check whether a no. is prime or not.
  - c) To generate Fibonacci series up to requested terms.

### Object as argument to function, function returning object

11. Write program-using class having class name Darray. Darray has pointer to pointer to integer as data member to implement double dimension dynamic array and provide following facilities:
  - a) Constructor to input values in array elements.
  - b) Input member function to get input in array element
  - c) Output member function to print element value
  - d) Add member function to perform matrix addition using objects.
  - e) Subtract member function to perform matrix subtraction using objects.
  - f) Multiply member function to perform matrix multiplication using objects
12. Write program to create class complex having data members to store real and imaginary part. Provide following facilities:
  - a) Add two complex no. using objects. b) Subtract two complexes no. using objects.
  - c) Multiply two complexes no. using objects. d) Divide two complex no. using objects.

### Friend Function

13. Create class Polar having data members radius and angle. It contains member functions for taking input in data members and member function for displaying value of data members. Class Polar contains declaration of friend function add which accepts two objects of class Polar and returns object of class Polar after addition. Test the class using main function and objects of class Polar.
14. Write program to create class distance having data members feet and inch (A single object will store distance in form such as 5 feet 3 inch). It contains member functions for taking input in data members and member function for displaying value of data members. Class Distance contains declaration of friend function add which accepts two objects of class Distance and returns object of class Distance after addition. Class Distance contains declaration of another friend function Subtract that accepts two objects of class Distance and returns object of class Distance after subtraction. Test the class using main function and objects of class Distance.
15. Write a program to create class Mother having data member to store salary of Mother, create another class Father having data member to store salary of Father. Write a friend function, which accepts objects of class Mother, and Father and prints Sum of Salary of Mother and Father objects.

### Friend Class

16. Write a program to create class Mother having data member to store salary of Mother, create another class Father having data member to store salary of Father. Declare class Father to be friend class of Mother. Write a member function in Father, which accepts object of class Mother and prints Sum of Salary of Mother and Father Objects. Create member function in each class to get input in data member and to display the value of data member.

### Static Data Member

17. Create a class Counter having a static data member, which keeps track of no. of objects created of type Counter. One static member function must be created to increase value of static data member

*Prateek*  
27/4/19

*Prateek*  
27/4/19

*Prateek*  
27-04-19

*Prateek*

*Prateek*  
27/4/19

*Prateek*  
27-04-2019

as the object is created. One static member function must be created to decrease value of static data member as the object is destroyed. One static member function must be created to display the current value of static data member. Use main function to test the class Counter.

#### STRUCTURE AND CLASS

18. Define structure student. Structure student has data members for storing name, rollno, name of three subjects and marks. Write member function to store and print data.

#### COPY CONSTRUCTOR, CONSTRUCTOR OVERLOADING, THIS POINTER, CONSTRUCTOR WITH DEFAULT ARGUMENT.

19. Write program to create a class Polar which has data member radius and angle, define overloaded constructor to initialize object and copy constructor to initialize one object by another existing object keep name of parameter of parameterized constructor same as data members. Test function of the program in main function.

20. Write program to create a class Polar which has data member radius and angle, use constructor with default arguments to avoid constructor overloading and copy constructor to initialize one object by another existing object keep name of parameter of parameterized constructor same as data members. Test functioning of the program in main function

#### FUNCTION OVERLOAD, REFERENCE VARIABLE, PARAMETER PASSING BY ADDRESS, STATIC FUNCTION

21. Write a class having name Calculate that uses static overloaded function to calculate area of circle, area of rectangle and area of triangle.

22. Write a class ArraySort that uses static overloaded function to sort an array of floats, an array of integers.

23. Write a program using class, which uses static overloaded function to swap two integers, two floats methods use reference variable.

24. Write a program using class, which uses static overloaded function to swap two integers; two floats methods use parameter passing by address.

#### STRING, POINTER, AND OPERATOR OVERLOADING

25. Create class String having pointer to character as data member and provide following Facilities:

- Constructor for initialization and memory allocation.
- Destructor for memory release.
- Overloaded operators + to add two string object.
- Overloaded operator = to assign one string object to other string object.
- Overloaded operator == to compare whether the two string objects are equal or not.
- Overloaded operator < to compare whether first-string object is less than second-string object.
- Overloaded operator > to compare whether first-string object is greater than second-string object or not.
- Overloaded operator <= to compare whether first string object is less than or equal to second string object or not
- Overloaded operator >= to compare whether first string object is greater than or equal to second string object.
- Overloaded operator != to compare whether first string object is not equal to second string object or not.
- Overloaded insertion and extraction operators for input in data member and display out put of data members.

26. Create a class Matrix having data member double dimension array of floats of size 3x3. Provide following facilities:

- Overloaded extraction operator for data input.
- Overloaded insertion operator for data output.
- Overloaded operator + for adding two matrix using objects.
- Overloaded operator - for subtracting two using matrix objects.
- Overloaded operator \* for multiplying two using matrix objects.

#### OPERATOR OVERLOADING WITH FRIEND FUNCTION

27. Create a class Polar having radius and angle as data members. Provide following facilities:

- Overloaded insertion and extraction operators for data input and display.

*Devendra Kish*  
27/4/19

*Ranjana*  
27/4/19

*Shree D*  
27/4/19

*Jha*

*Bah*  
27/4/19

*Sher*  
27/4/19

- b) Overloaded constructor for initialization of data members.
  - c) Overloaded operator + to add two polar co-ordinates using objects of class Polar.
28. Create class DegreeCelsius having a single data member to hold value of temperature in degree Celsius. Provide following facilities:
- a) Overloaded operator ++ which will increase value of data member by 1 (consider postfix and prefix operator overloading).
  - b) Overloaded operator -- which will decrease value of data member by 1 (consider postfix and prefix operator overloading).
  - c) Overloaded insertion and extraction operators for input in data member and display value of data member.

#### OPERATOR OVERLOADING AND DATA TYPE CONVERSION

29. Create a class Polar that contains data member radius and angle. Create another class Cartesian in the same program and provide following facilities:
- a) It should be possible to assign object of polar class to object of Cartesian class.
  - b) It should be possible to assign object of Cartesian class to object of polar class.
30. Create a class Fahrenheit that contains a data member to hold temperature in Fahrenheit. Create another class Celsius that contains a data member to hold temperature in Degree Celsius; in the same program and provide following facilities:
- a) It should be possible to assign object of Fahrenheit class to object of Celsius class.
  - b) It should be possible to assign object of Celsius class to object of Fahrenheit class.
  - c) It should be possible to compare objects of class Fahrenheit and Celsius to find out which object contains higher temperature.

#### VOID POINTER, POINTER AND POINTER TO OBJECT

31. Create a program having pointer to void to store address of integer variable then print value of integer variable using pointer to void. Perform the same operation for float variable.
32. Write program to find biggest number among three numbers using pointer and function.
33. Write swapping program to demonstrate call by value, call by address and call by reference in a single program.
34. Write program to Create a class Employee having data members to store name of employee, employee id, salary. Provide member function for data input, output. Use Pointer to object to simulate array of object to store information of 3 employees and test the program in function main.

#### INLINE FUNCTION.

35. Write a program using inline function to calculate area of circle.
36. Write a program using inline function to find minimum of two functions. The inline function should take two arguments and should return the minimum value.

#### INHERITANCE

37. Create a class account that stores customer name, account number and type of account. From this derive the classes cur\_acct and sav\_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks:
- a) Accept deposit from customer.
  - b) Display the balance
  - c) Computer and deposit interest.
  - d) Permit withdrawal and update the balance.
  - e) Check for the minimum balance, impose penalty, necessary and update the balance.
38. Create a class circle with data member radius; provide member function to calculate area. Derive a class sphere from class circle; provide member function to calculate volume. Derive class cylinder from class sphere with additional data member for height and member function to calculate volume.
39. Consider an example of declaring the examination result. Design three classes:- student, exam and result. The student class has data members such as that representing roll number, name of student. Create the class exam, which contains data members representing name of subject, minimum

Anis 27/4/19
   
 Bujang 27/4/19
   
 Date
   
 Jee
   
 Bah 27/4/19
   
 Suresh 27-04-19

marks, maximum marks, obtained marks for three subjects. Derive class result from both student and exam classes. Test the result class in main function.

**VIRTUAL AND PURE VIRTUAL FUNCTION**

40. Create a base class shape having two data members with two-member function getdata (pure virtual function) and printarea (not pure virtual function). Derive classes triangle and rectangle from class shape and redefine member function printarea in both classes triangle and rectangle and test the functioning of classes using pointer to base class objects and normal objects.

Arun  
27/4/19

Benjamin  
27/4/19

Good  
27-04-19

Dee  
27/4/19

Bah  
27/4/19

Rush  
27-04-2019

**PRACTICAL WORK**  
**BCA-208 - LAB V: Database Management System Lab**

**1 Scheme of Examination:-**

Practical examination will be of 3 hours duration. The distribution of practical marks will be as follows

Program 1 (Oracle)	-	20
Program 2 (Oracle)	-	20
Program 3 (Oracle)	-	20
Viva	-	25
[ Practical Copy + Practical Sessional]	-	15
 Total	-	 100

- 2 In every program there should be comment for each coded line or block of code
- 3 practical files should contain printed programs with name of author, date, path of program, unit no. and printed output.
- 4 All the following programs or a similar type of programs should be prepared

**List of Practical**

1. Using the following database,

- Colleges (cname, city, address, phone, afdate)
- Staffs ( sid, sname, saddress, contacts)
- StaffJoins ( sid, cname, dept, DOJ, post, salary)
- Teachings ( sid, class, paperid, fsession, tsession)
- Subjects ( paperid, subject, paperno, papename)

Write SQL statements for the following –

- a. Create the above tables with the given specifications and constraints.
- b. Insert about 10 rows as are appropriate to solve the following queries.
- c. List the names of the teachers teaching computer subjects.
- d. List the names and cities of all staff working in your college.
- e. List the names and cities of all staff working in your college who earn more than 15,000
- f. Find the staffs whose names start with 'M' or 'R' and ends with 'A' and/or 7 characters long.
- g. Find the staffs whose date of joining is 2005.
- h. Modify the database so that staff N1 now works in C2 College.
- i. List the names of subjects, which T1 teaches in this session or all sessions.
- j. Find the classes that T1 do not teach at present session.
  - a. Find the colleges who have most number of staffs.
  - b. Find the staffs that earn a higher salary who earn greater than average salary of their college.
  - c. Find the colleges whose average salary is more than average salary of C2
  - d. Find the college that has the smallest payroll.
  - e. Find the colleges where the total salary is greater than the average salary of all colleges.
  - f. List maximum, average, minimum salary of each college
- a. List the names of the teachers, departments teaching in more than one department.
- b. Acquire details of staffs by name in a college or each college.
- c. Find the names of staff that earn more than each staff of C2 College.
- d. Give all principals a 10% rise in salary unless their salary becomes greater than 20,000 in such case give 5% rise.
- e. Find all staff that do not work in same cities as the colleges they work.

*Aravind*  
27/4/19

*Pranav*  
27/4/19

*Shoban*  
27-04-19

*Sanu*

*Bah*  
27/4/19

*Shan*  
27-04-19

- f. List names of employees in ascending order according to salary who are working in your college or all colleges.
- Create a view having fields sname, cname, dept, DOJ, and post
  - Create a view consisting of cname, average salary and total salary of all staff in that college.
  - Select the colleges having highest and lowest average salary using above views.
  - List the staff names of a department using above views.
2. Create the following database,
- Enrollment (enrollno, name, gender, DOB, address, phone)  
 Admission (admno, enrollno, course, yearsem, date, cname)  
 Colleges (cname, city, address, phone, afdate)  
 FeeStructure (course, yearsem, fee)  
 Payment (billno, admno, amount, pdate, purpose)
- Create the above tables with the given specifications and constraints.
  - Insert about 10 rows as are appropriate to solve the following queries.
  - Get full detail of all students who took admission this year class wise
  - Get detail of students who took admission in Bhilai colleges.
  - Calculate the total amount of fees collected in this session
    - By your college
    - by each college
    - by all colleges
    - List the students who have not payed full fee
      - in your college
      - in all colleges
    - List the number of admissions in your class in every year.
    - List the students in the session who are not in the colleges in the same city as they live in.
    - List the students in colleges in your city and also live in your city.
3. Create the following database,
- Subjects (paperid, subject, paper, papename)  
 Test (paperid, date, time, max, min)  
 Score (rollno, paperid, marks, attendance)  
 Students (admno, rollno, class, yearsem)
- Create the above tables with the given specifications and constraints.
  - Insert about 10 rows as are appropriate to solve the following queries.
  - List the students who were present in a paper of a subject.
  - List all roll numbers who have passed in first division.
  - List all students in BCA-II who have scored higher than average
    - in your college
    - in every college
  - List the highest score, average and minimum score in BCA-II
    - in your college
    - in every college
4. Using the following database
- Colleges (cname, city, address, phone, afdate)  
 Staffs (sid, sname, saddress, contacts)  
 StaffJoins (sid, cname, dept, DOJ, post, salary)  
 Teachings (sid, class, paperid, fsession, tsession)  
 Subjects (paperid, subject, paperno, papename)
- Write SQL statements for the following –
- Create the above tables with the given specifications and constraints.
  - Insert about 10 rows as are appropriate to solve the following queries.
  - List the names of the teachers teaching computer subjects.
  - List the names and cities of all staff working in your college.
  - List the names and cities of all staff working in your college who earn more than 15,000
5. Using the following database

*Dute*  
27/4/19

*Pourya*  
27/4/19

*2007*  
27-04-19

*27/4/19*

*Brak*  
27/4/19

*Suresh*  
27-04-19

Colleges (cname, city, address, phone, afdate)  
 Staffs (sid, sname, saddress, contacts)  
 StaffJoins ( sid, cname, dept, DOJ, post, salary)  
 Teachings ( sid, class, paperid, fsession, tsession)  
 Subjects ( paperid, subject, paperno, papername)

- Find the staffs whose names start with 'M' or 'R' and ends with 'A' and/or 7 characters long.
- Find the staffs whose date of joining is 2005.
- Modify the database so that staff N1 now works in C2 college.
- List the names of subjects which T1 teaches in this session or all sessions.

6. Using the following database

Colleges (cname, city, address, phone, afdate)  
 Staffs (sid, sname, saddress, contacts)  
 StaffJoins ( sid, cname, dept, DOJ, post, salary)  
 Teachings ( sid, class, paperid, fsession, tsession)  
 Subjects ( paperid, subject, paperno, papername)

- Find the classes that T1 do not teach at present session.
- Find the college who have most number of staffs.
- Find the staffs who earn a higher salary who earn greater than average salary of their college.
- Find the colleges whose average salary is more than average salary of C2
- Find the college that has the smallest payroll.
- Find the colleges where the total salary is greater than the average salary of all colleges.
- List maximum, average, minimum salary of each college

7. Using the following database

Colleges (cname, city, address, phone, afdate)  
 Staffs (sid, sname, saddress, contacts)  
 StaffJoins ( sid, cname, dept, DOJ, post, salary)  
 Teachings ( sid, class, paperid, fsession, tsession)  
 Subjects ( paperid, subject, paperno, papername)

- Find the classes that T1 do not teach at present session.
- List the names of the teachers, departments teaching in more than one departments.
- Acquire details of staffs by name in a college or each college.
- Find the names of staff who earn more than each staff of C2 college.
- Give all principals a 10% rise in salary unless their salary becomes greater than 20,000 in such case give 5% rise.
- Find all staff who donot work in same cities as the colleges they work.
- List names of employees in ascending order according to salary who are working in your college or all colleges.

8. Using the following database

Colleges (cname, city, address, phone, afdate)  
 Staffs (sid, sname, saddress, contacts)  
 StaffJoins ( sid, cname, dept, DOJ, post, salary)  
 Teachings ( sid, class, paperid, fsession, tsession)  
 Subjects ( paperid, subject, paperno, papername)

- Find the classes that T1 do not teach at present session.
- Create a view having fields sname, cname, dept, DOJ, and post
- Create a view consisting of cname, average salary and total salary of all staff in that college.
- Select the colleges having highest and lowest average salary using above views.
- List the staff names of a department using above views.

*Praveen*  
27/4/19

*Praveen*  
27/4/19

*Praveen*  
27-4-19

*Praveen*

*Praveen*  
27/4/19

*Praveen*  
27-4-19



9. Enrollment (enrollno, name, gender, DOB, address, phone)  
 Admission (admno, enrollno, course, yearsem, date, cname)  
 Colleges (cname, city, address, phone, afdate)  
 FeeStructure (course, yearsem, fee)  
 Payment (billno, admno, amount, pdate, purpose)
- Create the above tables with the given specifications and constraints.
  - Insert about 10 rows as are appropriate to solve the following queries.
  - Get full detail of all students who took admission this year classwise
  - Get detail of students who took admission in Bhilai colleges.
  - Calculate the total amount of fees collected in this session
    - by your college
    - by each college
    - by all colleges
10. Enrollment (enrollno, name, gender, DOB, address, phone)  
 Admission (admno, enrollno, course, yearsem, date, cname)  
 Colleges (cname, city, address, phone, afdate)  
 FeeStructure (course, yearsem, fee)  
 Payment (billno, admno, amount, pdate, purpose)
- List the students who have not payed full fee
    - in your college
    - in all colleges
  - List the number of admissions in your class in every year.
  - List the students in the session who are not in the colleges in the same city as they live in.
  - List the students in colleges in your city and also live in your city.
11. Subjects (paperid, subject, paper, papername)  
 Test (paperid, date, time, max, min)  
 Score (rollno, paperid, marks, attendance)  
 Students (admno, rollno, class, yearsem)
- Create the above tables with the given specifications and constraints.
  - Insert about 10 rows as are appropriate to solve the following queries.
  - List the students who were present in a paper of a subject.
  - List all roll numbers who have passed in first division.
  - List all students in BCA-II who have scored higher than average
    - in your college
    - in every college
  - List the highest score, average and minimum score in BCA-II
    - in your college
    - in every college

*Krin*  
27/4/19

*Ramya*  
27/4/19

*Arif*  
27-04-19

*Arif*  
27-4-19

*Blak*  
27/4/19

*Shun*  
27-04-2019

**PRACTICAL WORK**  
**BCA-209 - LAB VI: Operating System Lab**

**Scheme of Examination:-**

1. Practical examination will be of 3 hours duration. The distribution of practical marks will be as follows

Program 1	-	20
Program 2	-	20
Program 3	-	20
Viva	-	25
[ Practical Copy + Internal Record ]	-	15
<b>Total</b>	-	<b>100</b>
2. In every program there should be comment for each coded line or block of code
3. Practical file should contain printed programs with name of author, date, path of program, unit no. and printed output.
4. All the following programs or a similar type of programs should be prepared

**List of Practical**

1. Change your shell environment – path, home, ifs, mail, ps1, ps2, term, logname
  - i) at commandline
  - ii) at shell level
  - iii) at login level
2. Change the wallpaper, screensaver in GNOME, KDE
3. Install Linux with following specifications – username, password, partitions for various directories such as /etc, /home, etc
4. Add a user and password, change the password
5. Add & remove a group
6. Create partitions on your disk.
7. Install and configure (i) printer (ii) scanner

***Using vi editor do the following exercises***

1. In a file
  - i) replace the words 'has' with 'has not '
  - ii) Locate n<sup>th</sup> character
  - iii) Sort lines 21 to 40
2. In a file copy/cut and paste following text-
  - i At i<sup>th</sup> line, n lines to j<sup>th</sup> line .
  - ii Yank a few words
  - iii Cut and paste n words to i<sup>th</sup> position in l<sup>th</sup> line
2. Open two files 'txtfile' and 'newfile' and copy/cut 5 lines from txtfile and paste them in newfile using vi editor.
3. Open 'txtfile' and copy/cut following and paste to the 'newfile'
  - i i<sup>th</sup> to the last line in it
5. Create macro
  - ii to paste your name at any position in the file.
  - iii to map the 1<sup>st</sup> function key to search for "loop" and copy into the buffer 'a' all text following it up to but not including the string "end".
  - iv to remove all leading spaces in a file
  - v to save and quit vi editor in input mode

**Write commands**

- i. List all files that match a class.

*Kris*  
27/4/19

*Bijay*  
27/4/19

*27-30-19*  
*27-4-19*

*Dak*  
27/4/19

*Suresh*  
27-4-19

- ii. List all files that do not match a class.
- iii. Change the file permissions
- iv. Configure or set characteristics of your terminal. Describe any 3.
- v. Display the lines in a file that contain a particular word.
- vi. Append the contents of two files in a file JABC.
- vii. Count the number of files in a directory.

Write shell programs

- i. Display all the users currently logged in detail with column headers.
- ii. List all files in current directory and save the list in a file ABC. Also save the contents of the files in ABC and display the contents in ABC in sorted order.
- iii. Sort the contents of a file ABC and save it in OABC.
- iv. Display all the users currently logged in detail with column headers.
- v. To save current date & time, number of files & directories in the current directory and contents of all the files to a single file NFL.
- vi. To input a number and test whether it is +ve, -ve or zero.
- vii. To test whether a filename is a regular file or a directory or of other type.
- viii. To list only the directories in current path.
- ix. To print the greatest of three numbers.
- x. To print 12 terms of Fibonacci series.
- xi. To display all users currently logged in & also check a particular user every 30 seconds until he logs in.
- xii. To save current date & time, number of files in the current directory and contents of all the files matching a pattern to a single file NPFL.
- xiii. To display particular messages depending on the weekday.
- xiv. To display common messages for following group of days-Monday & Wednesday, Tuesday & Thursday and Friday & Saturday and other day.
- xv. To accept a string from the terminal and echo a suitable message if it doesn't have at least 9 characters.
- xvi. Write a Shell Script to find the factorial of a number.
- xvii. Write a Shell Script to swap two numbers using third variable.
- xviii. Write a Shell Script to print prime numbers between 1 to 20.
- xix. Write a Shell Script to greatest of three numbers.
- xx. Write a Shell Script to sort the contents of a file XYZ and save it in BCAII
- xxi. Write a Shell Script to display mathematical table of any number in the format Ex :-3\*1=3.

*Shin*  
27/4/19

*Pranjan*  
27/4/19

*Asad*  
27.4.19

*Prateek*  
27.4.19

*Prab*  
27/4/19

*Pranav*  
27-04-2019