

Printed Pages – 6

Roll No. :

B022313(022)

B.Tech. (Third Semester) Examination

Nov.-Dec. 2020

(AICTE Scheme)

(Computer Science and Engg. Branch)

PRINCIPLES of PROGRAMMING LANGUAGES

Time Allowed : Three hours

Maximum Marks : 100

Minimum Pass Marks : 35

Note : Attempt all questions. Part (a) is of 4 marks and Part (b), (c) and (d) are of 8 marks. Part (a) is compulsory and attempt any two parts from (b), (c) and (d). Assume any missing data. Use diagrams wherever required.

1. (a) True / False?

(i) Full form of DFD is Data Flow Diagram.

(ii) The flowchart depicts flow of control in program modules.

[2]

- (iii) We should try to minimize coupling.
- (iv) We should try to maximize cohesion.

(b) Explain :

- (i) Operational Acceptance Testing
- (ii) Alpha Testing
- (iii) Beta Testing
- (iv) Software Maintenance phase

(c) Explain following points about DFD :

- (i) Definition, purpose and symbols used.
- (ii) Purpose of Level-0 and Level-1 DFD

(d) Explain about :

- (i) Abstraction
- (ii) Refinement
- (iii) Modularity
- (iv) Software Architecture

2. (a) True / False?

- (i) Relative address is a memory address that represents some distance from a starting point or the base address.

[3]

(ii) The variable names, constants and instruction labels are the basic elements of the symbolic address space.

(iii) In late binding the calling of the function gets bound with the definition of the function at compile time.

(iv) In early binding the binding of the function call to the respective definition during runtime.

(b) Answer the following :

- (i) What is the need of Assembler?
- (ii) Compare Compiler and Interpreter.

(c) Answer the following :

- (i) Draw that diagram of Water fall model, and Prototyping model.
- (ii) Compare Water fall model and Prototyping model.

(d) Answer the following :

- (i) Draw the Scrum Development Model.
- (ii) Explain the components of Scrum Development Model.

[4]

3. (a) True / False?

- (i) Java is an example of Imperative programming language.
- (ii) Java is an example of Functional programming language.
- (iii) LISP is an example of Imperative programming language.
- (iv) LISP is an example of Functional programming language.

(b) Answer the following :

- (i) Explain Imperative programming
- (ii) Explain Functional programming

(c) Explain following about LISP :

- (i) Executor
- (ii) Atoms
- (iii) Lists
- (iv) Strings

(d) Explain following about PROLOG :

- (i) Variable and Atom

[5]

(ii) Clause or rule

4. (a) True / False?

- (i) C++ is an object oriented programming language.
- (ii) C++ is a procedural programming language.
- (iii) Objects are created using class.
- (iv) Objects are created using functions.

(b) Answer the following :

- (i) What are Static members in a class?
- (ii) Write a program in C++ to demonstrate the use of static members in a class.

(c) Answer the following :

- (i) What are Nested class?
- (ii) Write a program to demonstrate the use of Nested class.

(d) Answer the following :

- (i) Compare New and Delete operators.
- (ii) Write a program in C++ to demonstrate the use of New and Delete operators.

5. (a) True / False?

- (i) Try-Catch are part of exception handling in C++.
- (ii) Throw-Try-Catch are part of exception handling in C++.
- (iii) During inheritance un-named instance of child class is created in parent class instance.
- (iv) Templates implement runtime polymorphism.

(b) Answer the following :

- (i) Explain function templates with code example in C++.
- (ii) Explain class templates with code example in C++.

(c) Answer the following :

- (i) Explain Abstract class in C++.
- (ii) Write a program to create an Abstract class in C++.

(d) Answer the following :

- (i) Explain Hybrid inheritance in C++.
- (ii) Write a program in C++ to demonstrate Hybrid inheritance.

B022313(022)

B.Tech. (Third Semester) Examination
Nov.-Dec. 2024

(New Scheme)

(Computer Science and Engg. Branch)

PRINCIPLES of PROGRAMMING LANGUAGES

Time Allowed : Three hours

Maximum Marks : 100

Minimum Pass Marks : 35

Note : Attempt all questions. Part (a) is of 4 marks and part (b), (c) and (d) are of 8 marks. Part (a) is compulsory and attempt any two parts from (b), (c) and (d).

Unit-I

1. (a) Compare top-down and bottom-up approaches in program design.

- (b) Discuss various modularization criteria for breaking down a program into modules. Provide examples to illustrate your points.
- (c) What is structured flowchart? Briefly explain the logic behind each step.
- (d) Explain the concept of levels of abstraction in program design and illustrate how it contributes to maintainability and flexibility. Provide an example.

Unit-II

2. (a) Compare major paradigms of programming languages. Also provide an example of each.
- (b) Discuss the challenges and benefits of developing new programming languages in an existing landscape dominated by established languages.
- (c) Explain the characteristics of programming languages.
- (d) Explain the factor of influencing the evaluation pf programming languages.

Unit-III

3. (a) Compare and contrast the main principles of functional and logic programming. Provide one

[3]

example of a functional and one example of a logic programming language.

- (b) Discuss the advantages and disadvantages of using LISP for symbolic computing and artificial intelligence applications. Provide specific examples of its use in these domains.
- (c) Explain the syntax and structure of PROLOG clauses and goals, including the use of variables predicates, and unification. How do they contribute to logic program execution.
- (d) Compare and contrast the expressive power and problem solving approaches of LISP and PROLOG. In which situations might one language be more suitable than the other.

Unit-IV

- ~~4~~ (a) Explain the principle of encapsulation and its benefits in protecting data and promoting information hiding within a class.
- ~~4~~ (b) Write a simple C++ program demonstrating the creation and use of a class with member variables, methods, and constructors. Briefly explain your code.

[4]

- (c) Compare the compile-time and run-time polymorphism mechanisms in C++. Provide examples of each type and discuss their respective applications.
- (d) Discuss the concept of message passing in object-oriented programming and its role in communication between objects.

Unit-V

5. ~~(a)~~ Differentiate between constructors and destructors in object-oriented programming.
- ~~(b)~~ Write a C++ program demonstrating constructor overloading with different parameter types.
- ~~(c)~~ Explain the concept of operator overloading in C++ and its benefits for customizing the behaviour of existing operators for user-defined objects. Provide an example of overloading the ‘+’ operator for adding two complex number.
- (d) Discuss the different types of inheritance available in C++ with example.

**B022313(022)**

**B.Tech. (Third Semester) Examination
April-May 2023**

(New Scheme)

(Computer Science and Engg. Branch)

PRINCIPLES of PROGRAMMING LANGUAGES*Time Allowed : Three hours**Maximum Marks : 100**Minimum Pass Marks : 35*

Note : Attempt all questions. Part (a) is of 4 marks and part (b), (c) and (d) are of 8 marks. Part (a) is compulsory and attempt any two parts from (b), (c) and (d).

Unit-I

- | | |
|---|---|
| 1. (a) What is a Program and Algorithm? | 4 |
| (b) What is Flowchart? Explain flowchart in detail with suitable diagram. | 8 |
| (c) What are the design technique for programming language? | 8 |

[2]

- (d) Write short notes on : (any **two**) 8
- (i) Unit test
 - (ii) Module test
 - (iii) Sub-system test

Unit-II

2. (a) Write characteristics of Programming Language. 4
- (b) Explain briefly about storage management and data binding. 8
- (c) What is difference between syntax and semantics of programming language? 8
- (d) Write short notes on : (any **two**) 8
- (i) Translator
 - (ii) Compiler
 - (iii) Linker & Loader

Unit-III

3. (a) What is function and pseudo code? 4
- (b) Comparison of functional and logical programming language. 8

[3]

- (c) LISP begin as a pure functional language but gradually acquired more imperative features, why? 8
- (d) What is Operators? Explain different types of operators. 8

Unit-IV

4. (a) Write comparisoin between new and delete operators. 4
- (b) Explace basic concepts of object oriented programming. 8
- (d) Write short notes on : (any **two**) 8
- (i) Nested class
 - (ii) Static member
 - (iii) C++ compiler
- (d) (i) Write a program which reads a string and then points to number of characters in this.
- (ii) What will be out of the following : 6+2=8
- ```
include <iostream.h>
void main ()
{
 int a[] = {18, 48, 28, 58, 38, 68};
 int i, * k,
```

```

k = a;
for (i = 0; i < 3; i++)
{
 (out << * K << " ");
 K++ = 2;
}
}

```

**Unit-V**

5. (a) What its Templates? Write the features of templates. 4
- (b) What are the friend function? Write a program to demonstrate the use of member function of one class as a friend of another class. 8
- (c) Write a program to C++ to read two number's from the keyboard and display larger value on the screen. 8
- (d) Write short notes on : 8
- (i) Constructor with types. Explain it.
  - (ii) How the destrator is importance?

Printed Pages – 3

Roll No. : .....

**B022313(022)**

**B.Tech. (Third Semester) Examination  
April-May 2022**

**(AICTE Scheme)  
(Computer Science & Engg. Branch)**

**PRINCIPLES of PROGRAMMING LANGUAGES**

*Time Allowed : Three hours*

*Maximum Marks : 100*

*Minimum Pass Marks : 35*

*Note : Attempt any two parts from (a), (b) and (c) of each question. All question carry equal marks.*

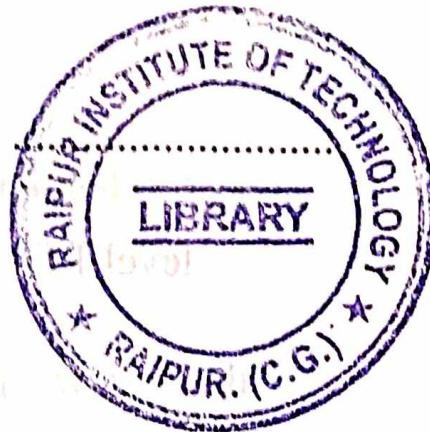
- |                                                      |    |
|------------------------------------------------------|----|
| 1. (a) Describe modules and modularization criteria. | 10 |
| (b) Define abstraction along with its levels.        | 10 |

[ 2 ]

- (c) Describe the test plan design guidelines. 10
2. (a) Describe the working principles of translators with an example. 10  
(b) Describe the various characteristics of programming language. 10  
(c) Write short notes on :  
    (i) Virtual computer  
    (ii) Binding and Binding times
3. (a) Differentiate between functional and imperative language. 10  
(b) How are list functions expressed in SML? Explain with suitable examples. 10  
(c) Explain Haskell programming language and its features. 10
4. (a) Differentiate between :  
    (i) Pointers and reference variables  
    (ii) New and Delete operators

[ 3 ]

- (b) Explain the basic structure of C++ program with an example. 10  
(c) Write a program in C++ to perform arithmetic operations. 10
5. (a) Explain the Friend function with an example. 10  
(b) Explain operator overloading with suitable example. 10  
(c) How exceptions are handled in C++ explain in brief? 10



**B022313(022)**

**B.Tech. (Third Semester) Examination**

**Nov.-Dec. 2022**

**(New Scheme)**

**(Computer Science and Engg. Branch)**

**PRINCIPLES of PROGRAMMING LANGUAGES**

***Time Allowed : Three hours***

***Maximum Marks : 100***

***Minimum Pass Marks : 35***

**Note :** Attempt all questions. Part (a) is of 4 marks and part (b), (c) and (d) are of 8 marks. Part (a) is compulsory and attempt any two parts from (b), (c) and (d). Assume any missing data. Use diagram wherever required.

- |                                                               |   |
|---------------------------------------------------------------|---|
| 1. (a) Define software testing and its types.                 | 4 |
| (b) Write about various program designing concepts in detail. | 8 |

[ 2 ]

[ 3 ]

(c) Explain DFD along with the purpose of level 0 and  
level 1 DFD.

8

(d) Describe in detail about various program design  
techniques.

8

2. (a) Write down the factors influencing evolution of a  
programming language.

4

(b) Explain various characteristics of a programming  
language in detail.

8

(c) Explain various storage management techniques and  
compare them.

8

(d) Describe various desirable features and design issues  
related with programming language processors.

8

3. (a) Explain AI in brief.

4

(b) Give detailed description about LISP.

8

(c) What is PROLOG? Explain in detail.

8

(d) Compare various functional and logic programming  
languages. Write their applications also.

8

4. (a) Write down various object oriented programming  
concepts.

4

(b) What is a static number? Write a program to  
demonstrate it.

8

(c) Write a C++ program to show the use of new and  
delete operations.

8

(d) Explain inheritance with a C++ program.

8

5. (a) Write short notes on constructor and destructor.

4

(b) Explain operator overloading with a program.

8

(c) Write a program in C++ to demonstrate the use of  
a friend function.

8

(d) Build a C++ program showing the use of virtual  
function.

8

## Printed Pages – 3

Roll No. : .....

B022313(022)

## **B.Tech. (Third Semester) Examination**

Nov.-Dec. 2021

(AICTE Scheme)

### **(Computer Science & Engg. Branch)**

# PRINCIPLES of PROGRAMMING LANGUAGES

**Time Allowed : Three hours.**

**Maximum Marks : 100**

### **Minimum Pass Marks : 35**

**Note:** Attempt all questions. Part (a) of each question is compulsory. Attempt any two parts from (b), (c) and (d). Part (a) carry 4 marks, part (b), (c) and (d) carry 8 marks.

1. (a) What are the modules and modularization criteria for programming language? 4

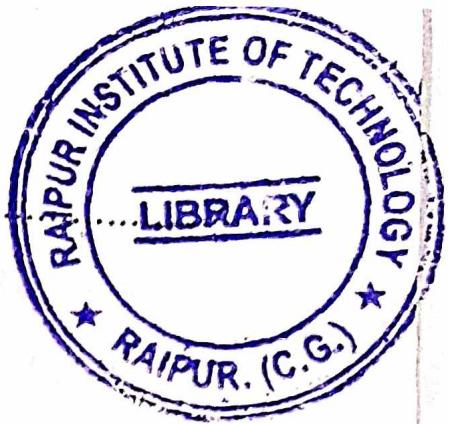
(b) Explain stepwise refinement design technique. 8

[ 2 ]

- (c) Explain Pseudo code and flow chart with example. 8
- (d) Define different levels of abstraction of programming language. 8
2. (a) Write down the important characteristics of programming language. 4
- (b) Explain structure and operations of translators in programming language. 8
- (c) Write down the factors influencing the evolution of programming language. 8
- (d) Explain Early binding and late binding in language. 8
3. (a) What is meant by symbolic expression in LISP? 4
- (b) Explain the programming structure for LISP. 8
- (c) Explain following : 8
- (i) Imperative Programming
- (ii) Functional Programming
- (d) Explain different datatypes available in PROLOG programming language. 8

[ 3 ]

4. (a) Explain following : 4
- (i) Abstraction
- (ii) Polymorphism
- (b) What is static member? Explain with example. 8
- (c) Explain this pointer with the help of example. 8
- (d) Write the differences between new and delete operators. 8
5. (a) Explain constructor and destructor in C++. 4
- (b) Explain following : 8
- (i) Friend function
- (ii) Abstract class
- (c) Explain virtual function with example. 8
- (d) Explain Exception Handling for a class with an example. 8



B022313(022)

**B.Tech. (Third Semester) Examination  
April-May 2024**

## (New Scheme)

## **(Computer Science and Engg. Branch)**

# PRINCIPLES of PROGRAMMING LANGUAGES

***Time Allowed : Three hours***

## **Maximum Marks : 100**

### **Minimum Pass Marks : 35**

**Note :** Attempt all questions. Part (a) is of 4 marks and part (b), (c) and (d) are of 8 marks. Part (a) is compulsory and attempt any two parts from (b), (c) and (d).

## **Unit-I**

1. (a) Define pseudo code with example. 4

(b) Explain module and modularization criteria in brief. 8

(c) What are the design techniques for programming language? 8

| 2 |

- (d) Write short notes on : 8
- (i) Structured flow chart
  - (ii) Decision table

**Unit-II**

2. (a) Explain virtual computer. 4  
 (b) Write and explain eight characteristics of programming language. 8  
 (c) Draw structure and explain the operation of translator. 8  
 (d) Explain syntax and semantics with its types. 8

**Unit-III**

3. (a) Explain imperative programming language. 4  
 (b) What is functional programming language? Explain its characteristics in brief. 8  
 (c) Explain LISP programming in terms of definition, control structure and functions. 8  
 (d) Explain PROLOG on the basis of lists and operators in brief. 8

| 3 |

**Unit-IV**

4. (a) Explain this pointer with example. 4  
 (b) Describe basic object oriented concept with example. 8  
 (d) Explain const and static keywords with suitable example. 8  
 (d) Explain basic structure of C++ programming with example. 8

**Unit-V**

5. (a) Explain destructor with example. 4  
 (b) What is friend function? Explain its characteristics with suitable example. 8  
 (c) What is operator overloading? Write a program to overload unary minus operator. 8  
 (d) Differentiate virtual function and pure virtual function with proper example. 8

Printed Pages – 3

Roll No. : .....

**B022313(022)**

**B.Tech. (Third Semester) Examination  
April-May 2021  
(AICTE Scheme)**

**(Computer Science & Engg. Branch)**

**PRINCIPLES of PROGRAMMING LANGUAGES**

**Time Allowed : Three hours**

**Maximum Marks : 100**

**Minimum Pass Marks : 35**

**Note : Attempt all questions. Part (a) of each question is compulsory and attempt any two parts from (b), (c) and (d) of each question.**

1. (a) What are test plans? Also write its types. 4

(b) What do you mean by modules and modularization criteria? 8

[ 2 ]

- (c) What are structured flow chart decisions? Do they differ from pseudo code? 8
- (d) Explain levels of abstraction. 8
2. (a) What are the characteristics of a good programming language? 4
- (b) Discuss the factors influencing the evolution of programming languages. 8
- (c) What are virtual computers? Also write its advantages. 8
- (d) Write short note on : Storage management. 8
3. (a) What are the applications of functional programming languages? 4
- (b) Compare : Functional and Logic programming language. 8
- (c) How do the control constructs in PROLOG function. 8
- (d) What are Lists? How do they operate? 8

[ 3 ]

4. (a) Compare new and delete operator. 4
- (b) What is this operator? Demonstrate its use with C++ program. 8
- (c) What are static members? Write a C++ program to explain the function of static member. 8
- (d) What are pointers? How do they differ from reference variables? 8
5. (a) Which classes are called abstract? 4
- (b) What is Dynamic binding? How does C++ implement it? 8
- (c) How does a private member of one class be accessed by another class? Explain with a program. 8
- (d) What are exceptions? How does C++ language handle exceptions? 8

Printed Pages – 3

Roll No. : .....



**B022313(022)**

**B.Tech. (Third Semester) Examination**  
**Nov.-Dec. 2023**

**(New Scheme)**

**(Computer Science and Engg. Branch)**

**PRINCIPLES of PROGRAMMING LANGUAGES**

***Time Allowed : Three hours***

***Maximum Marks : 100***

***Minimum Pass Marks : 35***

***Note : Attempt all questions. Part (a) is of 4 marks and part (b), (c) and (d) are of 8 marks. Part (a) is compulsory and attempt any two parts from (b), (c) and (d).***

### **Unit-I**

- 1. (a) Define external design and explain it in brief.**
- (b) Explain the fundamentals of design concepts in detail.**
- (c) Define modularity. Explain cohesion in detail with a suitable diagram.**

[ 2 ]

- (d) What do you mean by design notation? Explain Pseudo code with an example.

### **Unit-II**

2. (a) Explain the operations of translator in brief.  
(b) Explain factors influencing the evolution of programming languages in detail.  
(c) What do you mean by binding? Explain compile-time binding and run-time binding in detail.  
(d) Write short notes on :  
    (i) Virtual Computer  
    (ii) Simulated Computer

### **Unit-III**

3. (a) Explain logic programming language in brief.  
(b) Explain the comparison & applications of functional programming language and logic programming language in detail.  
(c) What is LISP? Explain the fundamentals of LISP.  
(d) What is PROLOG? Explain control constructs of PROLOG with examples.

[ 3 ]

### **Unit-IV**

4. (a) What do you mean by object? Explain Encapsulation in brief with an example.  
(b) Explain the features of object-oriented programming language in detail.  
(c) What is the basic structure of the C++ program?  
    Write a simple C++ program that demonstrates class and object.  
(d) What do you mean by pointer? Write a comparison between the pointer and reference variable.

### **Unit-V**

5. (a) Define the constructor and explain the constructor brief.  
(b) What do you mean by constructor? Explain function overloading with a suitable program.  
(c) Define friend function. Write a C++ program that demonstrates friend function and class.  
(d) Write a short note on :  
    (i) Abstract class  
    (ii) Inheritance

## **Chhattisgarh Swami Vivekananda Technical University, Bhilai**

|                                        |                                                                                                                                                               |
|----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Name of program:</b> Bachelor of    | <b>Semester:</b> III<br><b>Code:</b> B000311(014)<br><b>Total Tutorial Periods:</b> 01<br><b>Assignments:</b> Two<br><b>Maximum Marks:</b> 100 <b>Minimum</b> |
| <b>Technology Branch:</b> All Branches |                                                                                                                                                               |
| <b>Subject:</b> Mathematics – III      |                                                                                                                                                               |
| <b>Total Theory Periods:</b> 03        |                                                                                                                                                               |
| <b>Class Tests:</b> Two (Minimum)      |                                                                                                                                                               |
| <b>ESE Duration:</b> Three Hours       |                                                                                                                                                               |
| <b>Marks:</b> 35                       |                                                                                                                                                               |

### **Course Objectives:**

1. To provide knowledge of Laplace transform of elementary functions including its properties and applications to solve ordinary differential equations.
2. To have thorough knowledge of partial differential equations which arise in mathematical descriptions of situations in engineering.
3. To study about a quantity that may take any of a given range of values that can't be predicted as it is but can be described in terms of their probability.
4. To provide a thorough understanding of interpolation and methods to solve ordinary differential equation.

**UNIT-I** Laplace transform: Definition, Transform of elementary functions, Properties of Laplace transform, Transform of derivatives & integrals, Multiplication by  $t^n$ , Division by t, Evaluation of integrals, Inverse Laplace Transform, Convolution theorem, Unit step function, Unit impulse function, Periodic function, Application to solution of ordinary differential equations.

**UNIT- II Partial differential equation:** Formation, Solution by direct integration method, Linear equation of first order, Homogeneous linear equation with constant coefficients, Non-homogeneous linear equations, Method of separation of variables.

**UNIT- III Random variable:** Discrete and continuous probability distributions, Mathematical expectation, Mean and Variance, Moments, Moment generating function, probability distribution, Binomial, Poisson and Normal distributions.

**UNIT- IV Interpolation with equal and unequal intervals:** Finite differences, Newton's Forward & Backward Difference Formulae, Central Difference Formula, Stirling's Formula, Bessel's Formula, Lagrange's Formula and Newton's Divided Difference Formula.

**UNIT-V Numerical Solution of Ordinary Differential Equations:** Picard's Method, Taylor's Series Method, Euler's Method, Euler's Modified Method, Runge-Kutta Methods, Predictor-corrector Methods- Milne's Method, Adams-Bashforth Method.

**Text Books:**

1. "Higher Engg. Mathematics", Dr. B.S. Grewal– Khanna Publishers.
2. "Advanced Engg. Mathematics" , Erwin Kreyszig – John Wiley & Sons.
3. "Numerical Methods in Engineering and Science" , Dr. B.S. Grewal, Khanna Publishers.
4. "Numerical Methods for Scientific and Engineering Computation" , M .K. Jain, S. R. K

**Reference Books:**

1. "Applied Mathematics", P. N. Wartikar& J. N. Wartikar. Vol-II Pune Vidyarthi Griha Prakashan, Pune.
2. "Applied Mathematics for Engineers & Physicists", Louis A. Pipes- TMH.
3. "Numerical Methods for Scientists and Engineers" K. Shankar Rao, Prentice Hall of India.
4. "Numerical Methods" P. Kandasamy, K. Thilagavathy and K. Gunavathi, S. Chand publication.

**Course outcomes:** After studying the contents of the syllabus in detail the students will be able to: Define (mathematically) unit step unit impulse, Laplace transform its properties, inverse and applications to solve ordinary differential equations and find Numerical solution of differential equations, which may be arising due to mathematical modelling based on engineering problems. Hands on these Mathematical topics will make them equipped to prepare for higher studies through competitive examinations.

## Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                        |                                                  |
|------------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                                | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Principles of Programming Languages</b>                    | Course Code: <b>B022313(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>100 / 35</b> | L: 2 T: 1 P: 0 Credits: 3                        |
| Class Tests & Assignments to be conducted: 2 each                      | Duration (End Semester Exam): <b>03 Hours</b>    |

**UNIT- I Program Design:** Introduction- fundamental design concepts - Modules and modularization criteria – Design notation: Procedure template, Pseudo code - Structured flow chart decision. Tables - Design techniques: Stepwise refinement, Levels of abstraction, Top down- Test Plans-Design guidelines.

**UNIT-II Programming language processors:** Characteristics of programming languages, Factors influencing the evolution of programming language, Development in programming methodologies, desirable features and design issues, Structure and operations of translators, software simulated computer, syntax, semantics, structure, virtual computers, binding and binding times, storage management comparisons.

**UNIT- III Functional & Logic programming languages:** Introduction, comparison and applications of functional and logic programming languages; fundamentals of LISP (Objects, Control constructs, List processing) & PROLOG (Syntax, Lists, Operators and arithmetic, Control constructs).

**UNIT-IV Object-Oriented Programming Concepts-I:** Introduction to Basic Object-Oriented Concepts: (Object, Class, Encapsulation, Abstraction, Data Hiding, Inheritance, Polymorphism, Message Passing), Basic structure of a C++ program, C++ Compiler, C++ Classes, Methods, Objects, Nested Class, Const, Static members, this pointer, Comparison between Pointer and Reference Variables, Comparison between New and Delete Operators.

**UNIT-V Object-Oriented Programming Concepts-II:** Constructor, Destructor, Function and Operator Overloading, Friend functions and Friend classes, Inheritance, Abstract classes, Polymorphism, Virtual Function and Classes, Dynamic Binding, Exception Handling and Templates.

### **Text Books:**

1. “Software Engineering Concepts” by Richard Fairley, Tata McGraw Hill,
2. “Programming Languages, Design and implementation” by Terrance W. Pratt, and Marvin V. Zelkowitz, Prentice-Hall of India, Fourth edition, 2002.
3. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill

### **Reference Books:**

1. “Programming Languages – Concepts and Constructs” by Ravi Sethi, Addison-Wesley, 2nd Ed. 1996.
2. “Programming Languages: Principles and Paradigms” by Allen B. Tucker, Robert Noonan, TMH, 2006.

### **Course Outcomes [After undergoing the course, students will be able to:]**

1. Obtain broad understanding of the role of computer science, fundamental software design concepts and notations.
2. Get an overview of various programming language paradigms, processors & software simulation types.
3. Understand key concepts in the implementation of common features of programming languages.
4. Acquire knowledge of basic concepts about object-oriented programming languages.
5. Program in object-oriented programming language paradigm using various computational methods.

# Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                        |                                                  |
|------------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                                | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Data Structures &amp; Algorithms</b>                       | Course Code: <b>B022312(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>100 / 35</b> | L: 3 T: 1 P: 0 Credits: 4                        |
| Class Tests & Assignments to be conducted: 2 each                      | Duration (End Semester Exam): <b>03 Hours</b>    |

**UNIT- I** Introduction: Basic Terminology, Elementary Data Organization, Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations: Big-Oh, Time-Space trade-off. Abstract Data Types (ADT) Arrays: Definition, Single and Multidimensional Arrays, Row Major & Column Major Order Representation of Arrays, Application of arrays, Sparse Matrices and their representations. Linked lists: Array Implementation and Dynamic Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition, Generalized Linked List.

**UNIT-II** Stacks: Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion, Tower of Hanoi Problem, Simulating Recursion, Principles of recursion, Tail recursion, Removal of recursion Queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.

**UNIT- III** Trees: Basic terminology, Binary Trees, Binary Tree Representation: Array Representation and Dynamic Representation, Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Array and Linked Representation of Binary trees, Search Trees: Binary Search Trees (BST), Insertion and Deletion in BST Trees, Traversal algorithms: Inorder, Preorder and Postorder, Threaded Binary trees, Traversing Threaded Binary trees.

**UNIT-IV** Graphs: Terminology, Sequential and linked Representations of Graphs: Adjacency Matrices, Adjacency List, Adjacency Multi list, Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Transistive Closure and Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm.

**UNIT-V** Searching: Sequential search, Binary Search, Comparison and Analysis Internal Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Two Way Merge Sort, Heap Sort, Radix Sort, Tree (BST) Sort; Complexity of Search Algorithm, AVL trees, Introduction to m-way Search Trees, B Trees & B+ Trees, Hashing: Hash Function, Collision Resolution Strategies, Storage Management: Garbage Collection and Compaction.

## **Text books:**

1. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein “Data Structures Using C and C/C++”, PHI
2. Horowitz and Sahani, “Fundamentals of Data Structures”, Galgotia Publication.
3. Lipschutz, “Data Structures” Schaum’s Outline Series, TMH

## **References books:**

1. Jean Paul Trembley and Paul G. Sorenson, “An Introduction to Data Structures with applications”, McGraw Hill
2. R. Kruse etal, “Data Structures and Program Design in C”, Pearson Education
3. G A V Pai, “Data Structures and Algorithms”, TMH

**Course Outcomes** [After undergoing the course, students will be able to:]

1. Have a comprehensive knowledge of the data structures and algorithms on which file structures and data bases are based.
2. Understand the importance of data and be able to identify the data requirements for an application.
3. Have in depth understanding and practical experience of algorithmic design and implementation.
4. Have practical experience of developing applications that utilize databases.
5. Understand the issues involved in algorithm complexity and performance.

## Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                        |                                                  |
|------------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                                | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Digital Electronics</b>                                    | Course Code: <b>B022314(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>100 / 35</b> | L: 3 T: 1 P: 0 Credits: 4                        |
| Class Tests & Assignments to be conducted: 2 each                      | Duration (End Semester Exam): <b>03 Hours</b>    |

**UNIT- I Digital Fundamentals:** Weighted & Non-weighted codes, Sequential codes, self-complementing codes, Cyclic codes, 8-4-2-1 BCD code, Excess-3 code, Gray code: Binary to Gray and Gray to binary code conversion, Error detecting code, Error correcting code, 7-bit Hamming code, ASCII code, Binary Arithmetic, Boolean Algebra, Minimization of Switching Function , Demorgan's Theorem, Karnaugh's Map Method, Quine-McCluskey's Method (Tabular Method). Basic and Universal logic Gates, Realization of switching functions using gates.

**UNIT-II Digital Logic Families and Memory:** Transistor Inverter: Basic Concepts of RTL and DTL; TTL: Open collector gates, TTL subfamilies, IIL, ECL; MOS Logic: CMOS Logic, Dynamic MOS Logic, Interfacing: TTL to ECL, ECL to TTL, TTL to CMOS, CMOS to TTL, and Comparison among various logic families. Memories: ROM and RAM, PLA, PAL and FPGA;

**UNIT- III Combinational Circuits:** Adder & Subtractor: Half adder, Full adder, Half-subtractor, Full-subtractor, Parallel Binary adder, Look Ahead carry adder, Serial adder, BCD adder. Code converter, Parity bit generator/Checker, Comparator. Decoder: 3-line to 8-line decoder, 8-4-2-1 BCD to Decimal decoder, BCD to Seven segment decoder. Encoder: Octal to binary and Decimal to BCD encoder. Multiplexer: 2-input multiplexer, 4-input multiplexer. De- multiplexer: 1-line to 4-line, study of Multiplexer as Universal Logic Function Generator.

**UNIT-IV Sequential Circuits:** Flip-Flops: SR, JK, T, D, Master/Slave JK FF and their conversion, Excitation Tables. Introduction to registers (SISO, SIPO, PIPO, PISO) and Design of Counters- Ripple Counters, Ring Counters, Shift registers, Universal Shift Register.

**UNIT-V Machines and Application:** Finite State Machine, Mealy Machine, Moore Machine, **Introduction to VHDL:** Behavioral – data flow and algorithmic and structural description, lexical elements, data objects types, attributes, operators; VHDL coding examples.

### **Text Books:**

1. R. P. Jain: "Modern Digital electronics", TMH.
2. B. Somanathan Nair, "Digital Electronics & Logic Design", Prentice-Hall of India.
3. Pedroni V.A., "Digital Circuit Design with VHDL", Prentice Hall, India 2<sup>nd</sup> Edition.

### **Reference Books:**

1. R J Tocci, "Digital System principles and Applications"
2. "Digital Electronics " by A.K.Maini, Wiley India.
3. M.M. Mano: "Digital logic and computer design", PHI.
4. Floyd: "Digital fundamentals", UBS.

### **Course Outcomes** [After undergoing the course, students will be able to:]

1. Apply digital coding concepts to simplify circuit design.

2. Analyze the operations of various logic families and different semiconductor memories.
3. Design and implement various combinational circuits
4. Outline the concepts of latch circuits, flip flops and counters.
5. Design and develop basic digital systems using VHDL.

# Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                        |                                                  |
|------------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                                | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Operating System</b>                                       | Course Code: <b>B022315(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>100 / 35</b> | L: 3 T: 1 P: 0 Credits: 4                        |
| Class Tests & Assignments to be conducted: 2 each                      | Duration (End Semester Exam): <b>03 Hours</b>    |

**UNIT- I Introduction:** Operation System objectives and functions, The Evolution of operating Systems, Batch Systems, interactive systems, time sharing and real time systems, Protection. Operating System Structure, System Components, operating system service, System structure. Distributed Computing.

**UNIT-II Concurrent Processes:** Process concept: Introduction, Definitions, Process States, Process State Transitions, The process Control Block, Operations on Processes, Suspend and Resume, Interrupt Processing. Mutual Exclusion, the Producer / Consumer problem, the critical section problem, Semaphores, Classical problems in concurrency, inter process communication. CPU scheduling: concepts, performance criteria, and scheduling Algorithms. Algorithm evaluation, Multiprocessor scheduling.

**UNIT- III Dead Locks:** System model, Deadlock characterization. Prevention, Avoidance and Detection, Recovery from deadlock, combined approach.

**UNIT-IV Memory Management:** Base machine, Resident Monitor, multiprogramming with fixed partition, Multiprogramming with variable partitions, Paging, Segmentation, paged - segmentation, virtual Memory concepts, Demand paging, performance, Page Replacement algorithms, Allocation of frames, Thrashing, cache memory organization.

**UNIT-V I/O Management & Disk Scheduling:** I/O system Interrupts Direct Memory Access, I/O Buffering, File system: File Concepts – File organization and Access mechanism, File Directories, File sharing, Implementation issues. Disk Scheduling algorithms. Case Study on LINUX: Kernel and Buffer Cache Architecture, concept of inode file & directory structure, Basic system calls (Open, Read, Write, namei, File and Record Close, File Creation, Creation of Special Files, Change Directory and Change Root, Change Owner and Change Mode

## Text Books:

1. Operating system concepts Galvin by Silberschatz, John Wiley & Sons
2. Operating System Design & Implementation by Tanenbaum, A.S., PHI.
3. The Design of Unix Operating System, Maurice J. Bach, Pearson Education.

## Reference Books :

1. Modern Operating System: Andrew S. Tanenbaum, PEARSON EDUCATION INTERNATIONAL
2. Operating System concepts by Silberschatz A and Peterson, J.L, PE- LPE.
3. Operating systems: Internals & Design Principles, William Stallings, PHI.

## Course Outcomes [After undergoing the course, students will be able to:]

1. Identify the role of operating system in making computers execute data-processing jobs.
2. Realize managing computer's resource complexity during concurrent process execution through OS layers.
3. Analyse the reasons of resource bottlenecks-concurrency, deadlock and various synchronization mechanisms available.
4. Understand the functioning of operating system components in Memory Management techniques, Virtual Memory Management.
5. Understand disk organization, file system structure, Secondary Storage Management functions of OS.

## Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                       |                                                  |
|-----------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                               | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Data Structures (Laboratory)</b>                          | Course Code: <b>B022321(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>40 / 20</b> | L: 0 T: 0 P: 2 Credit(s): 1                      |

***List of Experiments: (At least 10 experiments are to be performed by each student)***

1. Write a program to perform following operations in one dimensional array, Insertion, Deletion and Searching (Linear & Binary).
2. Write a program to implement stack and perform push and pop operations.
3. Write a program to convert infix to postfix expressions using stack.
4. Write a program to perform following operations on a linear queue - addition, deletion, traversing.
5. Write a program to perform following operations on a circular queue - addition, deletion, traversing.
6. Write a program to perform following operations on a double ended queue - addition, deletion, traversing.
7. Write a program to perform following operations on a single link list-creation, insertion, deletion.
8. Write a program to perform following operations on a double link list – creation, insertion, deletion.
9. Write a program to implement polynomial in link list and perform.
  - a) Polynomial arithmetic
  - b) Evaluation of polynomial
10. Write a program to implement a linked stack and linked queue.
11. Write programs to perform Insertion, selection and bubble sort.
12. Write a program to perform quick sort.
13. Write a program to perform merge sort.
14. Write a program to perform heap sort.
15. Write a program to create a Binary search tree and perform –insertion, deletion & traversal.
16. Write a program to traversal of graph (Breadth-first Search, Depth-first Search methods)

**Remarks:** The students are free to choose any programming platform from (C++ / JAVA / PYTHON) to perform the above-mentioned set of laboratory experiments.

**Laboratory Outcomes** [After undergoing the course, students will be able to:]

1. Understand the importance of abstract data types, structure types and their usability in different applications through different programming platforms.
2. Implement various data structure operations (traversal, accession, insertion, deletion & updation) on stacks, linked lists, queues, trees & graphs.
3. Design and analyse the time and space efficiency of implemented data structures
4. Identify the selection of appropriate data structure for given problem situations.
5. Implement various kinds of searching and sorting techniques.

**Recommended Books:**

1. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein “Data Structures Using C and C/C++”, PHI
2. Jean Paul Trembley and Paul G. Sorenson, “An Introduction to Data Structures with applications”, McGraw Hill.
3. Lipschutz, “Data Structures” Schaum’s Outline Series, TMH

## Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                       |                                                  |
|-----------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                               | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Digital Electronics Laboratory</b>                        | Course Code: <b>B022322(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>40 / 20</b> | L: 0 T: 0 P: 2 Credit(s): 1                      |

***List of Experiments: (At least 10 experiments are to be performed by each student)***

1. To study the characteristics and operations of TTL Inverters, OR, AND, NOR and NAND gate using ICs.
2. To study NAND and NOR gates as a universal logic.
3. To study and prove Demorgan's Theorem .
4. To design Half and Full adder circuits using logic gates.
5. To design Half and full subtractor circuits using logic gates.
6. To study the binary parallel adder.
7. To design 4 bit magnitude comparator circuits.
8. To study the 7 segment decoder .
9. To design 4:16 decoder using two 3:8 decoder and four 2:4 decoder
10. To design 16: 1 Multiplexer using 4:1 Multiplexer.
11. To study various types of flip flops using logic gates and ICs.
12. To design Mode-N and divide by K counter.
13. To construct a 4 bit binary to gray converter and vice versa using IC 7486 .
14. To study Up-Down counter.
15. To study programmable shift registers.

***Experiments using VHDL (At least 4 Experiments are to be performed by each student)***

1. Design AND,OR,XOR gates.
2. Design Half Adder (Data Flow Style)
3. Design Half Adder (Behavioural Style)
4. Design Half Adder (Structural style Direct entity instantiation)
5. Design Half Adder (Structural style indirect entity instantiation(Component))
6. Design Half Adder (Mixed Style)
7. Design 4 bit comparator Using std\_logic\_vector inputs.
8. Design 4:1 Multiplexer using Boolean expression
9. Design the 7 segment decoder .
10. Design 3:8 decoder

**Laboratory Equipment / Machine Requirements:** Logic gate trainer, Digital ICs Trainer, Various ICs 7400,7402,7404,7408,7432,7486,74138,74151,74155 etc, Xilinx ISE WebPACK

**Laboratory Outcomes** [After undergoing the course, students will be able to:]

1. Acknowledge about the fundamentals of digital circuit Design.
2. Understand the concepts of logic families.
3. Take interest to design and develop ICs in VLSI industries.
4. Understand the operations of latch circuits, flip flops, counters & semiconductor memories.
5. Understand and design combinational circuits.

**Recommended Books:**

1. M.M. Mano : "Digital Logic and Computer Design";
2. Kenneth L. SHORT "VHDL FOR ENGINEERS", Pearson Education.

## Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                       |                                                  |
|-----------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                               | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Operating Systems (UNIX) Laboratory</b>                   | Course Code: <b>B022323(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>40 / 20</b> | L: 0 T: 0 P: 2 Credit(s): 1                      |

***List of Experiments: (At least 10 experiments are to be performed by each student)***

1. Practice session: Study the features of Linux environment, basic Linux commands (echo, who, date, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, find, tail, head, sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, comm, cmp, diff, tr); also document the syntax and semantics of those commands.
2. Write a shell script that accepts a name from the user and displays whether it is a file, directory or something else.
3. Write a shell script that creates users; also check if a particular user has logged in or not. If not, continue the loop till he/she logins. Once the required user logins, display a message.
4. Write a shell script that searches for a given string in a text input file.
5. Write a shell script that compiles all C files in your home directory and creates executable files.
6. Write a shell script that given a filename as argument, deletes all even lines in a file & removes duplicate lines from a file.
7. Write a shell script that enhances find command by adding error messages that explain why the command failed.
8. Write a shell script to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer. Calculate percentage and grade according to following logic:

Percentage  $\geq 90\%$  : Grade A, Percentage  $\geq 80\%$  : Grade B, Percentage  $\geq 70\%$  : Grade C, Percentage  $\geq 60\%$  : Grade D, Percentage  $\geq 40\%$  : Grade E, Percentage  $< 40\%$  : Grade F.

9. Write a shell script to accept the name, grade, and basic salary from the user. Write the details into a file called employee, separating the fields with a colon (:) continue the process till the user wants.
10. Write an Awk' script to count the number of lines in a file that do not contain vowels.
11. Write an Awk' script to find the number of characters, words and lines in a file.
12. Write a C program to simulate following non-preemptive CPU scheduling algorithms: a) FCFS b) SJF c) Round Robin (pre-emptive) d) Priority scheduling techniques.
  - a. to find average turnaround times and waiting times;
  - b. to display / print Gantt Chart (in any convenient format).
13. Implement the Producer – Consumer problem using semaphores (using UNIX system calls).
14. Write a C program to simulate disk-scheduling algorithms: a) FCFS b) SCAN c) C-SCAN techniques.
15. Write a C program to simulate page replacement algorithms: a) FIFO b) LRU c) LFU d) OPT techniques.

**Remarks:** The laboratory experiments may be performed in with of the LINUX shell environments: BOURNE Shell / KORN Shell / 'C' Shell.

**Laboratory Outcomes** [After undergoing the course, students will be able to:]

1. Understand the concept of Unix and shell programming.
2. Learn the working of Linux OS Kernel.
3. Analyse the differences between features provided in Windows and Linux operating system.
4. Learn the concept of loops and decision-making statements.
5. Analyse the logic & procedure of problem solving through Scripts.

**Recommended Books:**

1. Advance UNIX, a Programmer's Guide, S. Prata, BPB Publications, New Delhi.
2. The Complete Reference Unix, Rosen, Host, Klee, Farber, Rosinski, Second Edition, TMH.

# Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                                       |                                                  |
|-----------------------------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>                               | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: <b>Software Laboratory (SciLAB / MATLAB)</b>                 | Course Code: <b>B022324(022)</b>                 |
| <b>Total / Minimum-Pass Marks</b> (End Semester Exam): <b>40 / 20</b> | L: 0 T: 0 P: 2 Credit(s): 1                      |

**List of Experiments:** (At least 10 experiments are to be performed by each student)

1. Identification of different matrix types.
2. Properties and Operations of arrays and matrices.
3. Write a program to find probability of tossing a coin and rolling a die through large no. of experimentation.
4. Compute y- coordinates of a STRAIGHT LINE  $y = mx + c$ , where slope of line  $m = 0.5$ , intercept  $c = -2$  and x-coordinates :  $x = 0$  to  $10$  for  $0.5$  increments.
5. Plot  $y = \sin x$  where  $0 \leq x \leq 2$ .
6. Plot  $y = e^{-0.4x} \sin x$  where  $0 \leq x \leq 4$ .
7. Find the solution of linear algebraic equations in 2 variables, 3 variables:
  - a.  $x + 4y = 18; 2x + 3y = 16$
  - b.  $x + 2y + 3z = 1; 3x + 3y + 4z = 1; 2x + 3y + 3z = 2$
8. Determination of roots of a given polynomial & quadratic equations.
9. Determination of Eigen Value & Eigen Vectors for matrices.
10. Write a script file to draw a unit circle.
11. Write a function factorial to compute the factorial  $n!$  for any integer  $n$ .
12. Write a function factorial to compute the factorial  $n!$  using RECURSION for any integer  $n$ .
13. Write a function to compute the geometric series  

$$1 + r + r^2 + r^3 + \dots + r^n$$
 for given  $r$  and  $n$ .
14. Write a function file *crossprod* to compute the cross product of two vectors  $u$  and  $v$ .
15. Design of a toy project as an independent study towards problem-based learning.

**Laboratory Equipment:** The experiments may be performed in FOSS (Spoken Tutorials SciLAB Project: [www.scilab.org](http://www.scilab.org), [www.scilab.in](http://www.scilab.in)).

**Laboratory Outcomes** [After undergoing the course, students will be able to:]

1. Understand the main features of the MATLAB/SCILAB program development environment to enable their usage in the higher learning.
2. Realize the power of interactive calculation, programming, graphics, animation in SciLAB / MATLAB and complete portability across platforms.
3. Enjoy SciLAB / MATLAB as a scientific computing and visualization tool.
4. Explore Interactive Computation with matrices and arrays of n-dimensions.
5. Interpret and visualize simple mathematical functions and operations there on using plots/display.

## **Recommended Books:**

1. Getting started with MATLAB: A Quick Introduction for Scientists and Engineers by Rudra Pratap, IIS Banaglore.
2. Scilab Manual for Probability Theory and Statistics Lab by Prof S N Chandra Shekhar; [https://scilab.in/lab\\_migration\\_run/82](https://scilab.in/lab_migration_run/82)
3. Scilab Manual for Numerical techniques lab by Prof Kanika Gupta; [https://scilab.in/lab\\_migration\\_run/82](https://scilab.in/lab_migration_run/82)
4. Scilab Manual for Probability Theory and Random Processes by Prof Shital Thakkar; [https://scilab.in/lab\\_migration\\_run/82](https://scilab.in/lab_migration_run/82)
5. Scilab Manual for Numerical Techniques by Dr Javed Dhillon; [https://scilab.in/lab\\_migration\\_run/82](https://scilab.in/lab_migration_run/82)

## Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

|                                                |                                                  |
|------------------------------------------------|--------------------------------------------------|
| Program / Semester: <b>B.Tech (III)</b>        | Branch: <b>Computer Science &amp;Engineering</b> |
| Subject: Soft Skills & Personality Development | Course Code: B000306(046)                        |
| Total Marks (Internal Assessment): 10          | L: 0 T:0 P: 2 Credit(s): 0                       |
| Internal Assessments to be conducted: 02       | Duration (End Semester Exam): NA                 |

**UNIT-1 Communication Skills-Basics:** Understanding the communicative environment, Listening: What to listen for and why, When to speak and how, Starting and sustaining a conversation, Presentation and Interaction, Common errors during communication, Humour in Communication.

**UNIT-2 Interpersonal communication:** Building Relationships, Understanding Group Dynamics- I, Emotional and Social Skills, Groups, Conflicts and their Resolution, Social Network, Media and Extending Our Identities

**UNIT- 3 Vocational skills:** Managing time: Planning and Goalsetting, managing stress: Types of Stress; Making best out of Stress, Resilience, Work-life balance, Applying soft-skills to workplace

**UNIT-4 Mindsets and Handling People:** Definitions and types of Mindset, Learning Mindset, Developing Growth Mindset, Types of People, How to say NO

**UNIT-5 Inner Development:** Motivating oneself, Persuasion, Survival Strategies, Negotiation, Leadership and motivating others, controlling anger, Gaining Power from Positive Thinking.

### Text Books:

1. Petes S. J., Francis. Soft Skills and Professional Communication. New Delhi: Tata McGraw-Hill Education, 2011.
2. Stein, Steven J. & Howard E. Book. The EQ Edge: Emotional Intelligence and Your Success. Canada: Wiley & Sons, 2006.
3. Dorch, Patricia. What Are Soft Skills? New York: Execu Dress Publisher, 2013.

### Reference Books:

- Kamin, Maxine. Soft Skills Revolution: A Guide for Connecting with Compassion for Trainers, Teams, and Leaders. Washington, DC: Pfeiffer & Company, 2013.
- Canfield, Jack. The Success Principles (TM) — 10<sup>th</sup> Anniversary Edition: How to get from Where You Are to Where You want to Be. New York Times. 2009.
- Peale Norman Vincent. The Power of Positive Thinking: 10 Traits for Maximum Result. Paperback Publication. 2011.
- Klaus, Peggy, Jane Rohman & Molly Hamaker. The Hard Truth about Soft Skills. London: Harper Collins E-books, 2007.

### Course Outcomes [After undergoing the course, students will be able to:]

1. Learn to listen actively to analyse audience and tailor the delivery accordingly.
2. Increase their awareness of communication behaviour by using propriety profiling tool.
3. Master three “As” of stressful situation: Avoid, Alter, Accept; to cope with stressors and create a plan to reduce or eliminate them.
4. Develop growth mindset and able to handle difficult person and situations successfully.
5. Develop technique of turning negativity into positivity and generate self-motivation skills.