

20CST31: INTRODUCTION TO C++ PROGRAMMING**Credits – 3****L:T:P :: 3:0:0****Sessional Marks: 30****University Exam Marks: 70**

Course Objectives

To expose the students to the following:

1. Object Oriented Programming concepts using the C++ language.
2. The principles of data abstraction, inheritance and polymorphism.
3. Virtual functions and polymorphism.
4. Formatted I/O and unformatted I/O.
5. The concept of exception handling.

Course Outcomes

After successful completion of course the student should be able to

- CO1** Describe the procedural and object-oriented paradigm with concepts of streams, classes, functions, data and objects.
- CO2** Understand dynamic memory management techniques using pointers, constructors, destructors, etc
- CO3** Illustrate the concept of function overloading, operator overloading, virtual functions and polymorphism.
- CO4** Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming.
- CO5** Demonstrate the use of various OOPs concepts with the help of programs

UNIT I

Object-Oriented Thinking: Different paradigms for problem solving, Need for OOP paradigm, differences between OOP and Procedure oriented programming, Overview of OOP concepts: Abstraction, Encapsulation, Inheritance and Polymorphism.

C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References. Flow control statement- if, switch, while, for, do, break, continue, goto statements. Functions - Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions. Dynamic memory allocation and de-allocation operators-new and delete, Preprocess or directives.

UNIT II

C++ Classes and Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

UNIT III

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class. Virtual Functions and Polymorphism: Static and Dynamic binding, virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

UNIT IV

C++ I/O: I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and Stringstreams, Overloading operators, Error handling during file operations, Formatted I/O.UNIT V

Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions.

Textbooks

1. Herbert Schildt, “The Complete Reference C++”, 4th Edition, Tata McGraw Hill, 2014.
2. Walter Savitch, –Problem solving with C++: The Object of Programming, 4th Edition, Pearson Education, 2002.

Reference Books

1. B. Stroutstrup, “The C++ Programming Language”, 3rd Edition, Pearson Education, 2000.
2. T. Gaddis, J. Walters and G. Muganda, “OOP in C++”, 7th Edition, Pearson Education, 2010.
3. R. Lafore, “Object Oriented Programming in C++”, 3rd Edition, Galigotia Publications Pvt Ltd, 2004.

Web References

1. <https://nptel.ac.in/courses/106/105/106105151/>
2. <https://nptel.ac.in/courses/106/101/106101208/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H	-	-	-	-	-	-	-	-	-	-	H	L	-
CO2	H	-	-	-	-	-	-	-	-	-	-	-	H	L	-
CO3	-	L	H	M	-	-	-	-	-	-	-	-	L	H	-
CO4	M	H	-	-	-	-	-	-	-	-	-	-	M	M	-
CO5	L	M	-	H	-	-	-	-	-	-	-	-	M	H	-

20CSP13: ADVANCED PROGRAMMING LAB**Credits – 3****L:T:P :: 0:0:6****Sessional Marks: 40****University Exam Marks: 60****Course Objectives**

To expose the students to the following:

1. Programming methodologies using C language.
2. Identify Object Oriented Programming concepts using the C++ language.
3. Programming methodologies using Java.
4. Develop skills to design and analyze simple linear and non-linear data structures.

Course Outcomes

After successful completion of course the student should be able to

- CO1** Formulate the algorithms for simple problems and write iterative as well as recursive programs.
- CO2** Develop programs with OOPS concepts.
- CO3** Solve complex problems using Java.
- CO4** Develop skills to design and analyse simple linear and non-linear data structures

List of C Programs

1. Write a C program to read in two integers and display one as a percentage of the other. Typically, your output should look like 20 is 50.00% of 40 assuming that the input numbers were 20 and 40. Display the percentage correct to 2 decimal places.
2. Take three coefficients (a, b, and c) of a Quadratic equation ($ax^2+bx+c=0$) as input and compute all possible roots. Implement a C program to output the possible roots for a given set of coefficients with appropriate messages.
3. Given as input three integers representing a date as day, month, year, print the number day, month and year for the next day's date. Typical input: -28 2 1992, Typical output: -Date following 28:02:1992 is 01:03:1992
4. Write program for students marks grading.
5. Implement a C program that takes an integer number as input, check whether it is PALINDROME or NOT and output the reverse of the same with suitable messages. Ex: Num: 2014, Reverse: 4102, Not a Palindrome.
6. Write a C program for the problem given below: Assume that the United States of America uses the following income tax code formula for their annual income: First US\$ 5000 of income: 0% tax Next US\$ 10,000 of income: 10% tax Next US\$ 20,000 of income: 15% tax. An amount above US\$ 35,000: 20% tax. For example, somebody earning US\$ 38,000 annually would owe US\$ $5000 \times 0.00 + 10,000 \times 0.10 + 20,000 \times 0.15 + 3,000 \times 0.20$, which comes to US\$ 4600. Write a program that uses a loop to input the income and calculate and report the owed tax amount. Make sure that your calculation is mathematically accurate and that truncation errors are eliminated.
7. Consider the following algorithm to generate a sequence of numbers. Start with an integer n . If n is even, divide by 2. If n is odd, multiply by 3 and add 1. Repeat this process with the new value of n , terminating when $n = 1$. For example, the following sequence of numbers will be generated for $n = 22$: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
8. Write a program to find all (a) Armstrong number in the range of 0 and 999 Hint: An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. For example, 371 is an Armstrong number since $3^3 + 7^3 + 1^3 = 371$. (b) check whether a given number is a perfect number or not. Hint: A positive integer n is called a perfect number if it is equal to the sum of all of its positive divisors, excluding n itself. For example, 6 is a perfect number, because 1, 2 and 3 are its proper

positive divisors and $1 + 2 + 3 = 6$. The next perfect number is $28 = 1 + 2 + 4 + 7 + 14$. The next perfect numbers are 496 and 8128.

9. Write a program to check whether given two numbers are amicable numbers or not. Hint: Amicable numbers are two numbers so related that the sum of the proper divisors of the one is equal to the other, unity being considered as a proper divisor but not the number itself. Such a pair is (220, 284); for the proper divisors of 220 are 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 and 110, of which the sum is 284; and the proper divisors of 284 are 1, 2, 4, 71, and 142, of which the sum is 220.
10. Write a C program to do the following computation by providing the option using the switch statement:
 - a. Add two matrices
 - b. Subtract two matrices
 - c. Multiply two matrices
 - d. Check if the given matrix is magic square or not.
 - e. Print the upper and lower triangle of the matrix.
 - f. Compute transpose of a matrix.
 - g. Find the inverse of a matrix.
11. Write a C functions to convert decimal to (a) binary (b) octal (c) hexadecimal.
12. Write a function for each of the following and a program to
 - a. Convert a given lowercase string to upper case string without using the inbuilt string function.
 - b. Count number of vowels, consonants and spaces in a given string.
 - c. Find the length of a character string
13. Write a C program that uses functions to perform the following operations: i. To insert a sub-string in to a given main string from a given position. ii. To delete n Characters from a given position in a given string.
14. Define a structure that will hold the data for a complex number. Using this structure, please write a program that will input two complex numbers and output the multiple of the two complex numbers. Use double variables to represent complex number components. Note: A complex number z is a number of the form $z = a + bi$ where a and b are real numbers. The term a is called the real part of z and b is called the imaginary part of z . The multiplication operation on complex numbers is defined as: $(a + bi) * (c + di) = (ac - bd) + (ad + bc)i$
15. Write a program to create a file, open it, type-in some characters and count the number of characters in a file.
16. Write a program that will input a person's first name, last name, SSN number and age and write the information to a data file. One person's information should be in a single line. Use the function `fprintf` to write to the data file. Accept the information and write the data within a loop. Your program should exit the loop when the word 'EXIT' is entered for the first name. Remember to close the file before terminating the program. Hint: Use the function `strcmp()` to compare two strings.

LIST OF C++ PROGRAMS

1. Define a class to represent a bank account which includes the following members as Data members: a) Name of the depositor b) Account Number c) Withdrawal amount d) Balance amount in the account
Member Functions: a) To assign initial values b) To deposit an amount c) To withdraw an amount after checking the balance d) To display name and balance.
2. Write the above program for handling n number of account holders using array of objects.
3. Write a C++ program to compute area of right-angle triangle, equilateral triangle, isosceles triangle using function overloading concept.
4. Write a C++ program to swap the values two integer members of different classes using friend function.
5. Define a class string and overload `==` to compare two strings and `+` operator for concatenation of two strings.

6. Consider an example of declaring the examination result. Design three classes student, exam and result. The student has data members such as roll no, name. Create the class exam by inheriting the student class. The exam class adds data members representing the marks scored in 5 subjects. Derive the result from exam-class and it has own data members like total, avg. Write the interactive program into model this relationship.

LIST OF JAVA PROGRAMS

1. Write program(s) on use of inheritance, preventing inheritance using final, abstract classes.
2. Write program(s) on dynamic binding, differentiating method overloading and overriding.
3. Write program(s) on ways of implementing interface.
4. Write a program to develop an applet that displays a simple message, for waving a Flag using Applets and Threads.
5. Write program(s) which uses the exception handling features of the language, creates exceptions and handles them properly, uses the predefined exceptions, and create own exceptions
6. Write program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it's not a duplicate of any number already read. Display the complete set of unique values input after the user enters each new value.
7. Write program(s) on creating multiple threads, assigning priority to threads, synchronizing threads, suspend and resume threads
8. Write a program to split a given text file into n parts. Name each part as the name of the original file followed by .part<n> where n is the sequence number of the part file.
9. Write a program to create a super class called Figure that receives the dimensions of two-dimensional objects. It also defines a method called area that computes the area of an
 - a. object. The program derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub classes override area () so that it returns the area of a rectangle and triangle respectively.
10. Write a program that creates three threads. First thread displays -Good Morning every one second, the second thread displays -Hello every two seconds and the third thread displays -Welcome! every three seconds
11. Design a simple calculator which performs all arithmetic operations. The interface should look like the calculator application of the operating system. Handle the exceptions if any.
12. Write a program to handle mouse events
13. Write a program to handle keyboard events
14. Write a program that allows conduction of object type examination containing multiple choice questions, and true/false questions. At the end of the examination when the user clicks a button the total marks have to be displayed in the form of the message.
15. Write a program that creates menu which appears similar to the menu of notepad application of the Microsoft windows or any editor of your choice.
16. Write a program that creates dialog box which is similar to the save dialog box of the Microsoft windows or any word processor of your choice.
17. Write a program that correctly implements producer consumer problem using the concept of inter thread communication
18. Write a program to find and replace pattern in a given file.

LIST OF DATA STRUCTURE PROGRAMS

1. Write a program that uses stack operations to convert a given infix expression into its postfix Equivalent, Implement the stack using an array
2. Write a program to reverse the elements in the stack using recursion.
3. Write a program to implement the simple Queue and circular operations

4. Write a program that uses functions to perform the following: a) Create a singly linked list of integers. b) Delete a given integer from the above linked list. c) Display the contents of the above list after deletion.
5. Write a program that uses functions to perform the following: a) Create a doubly linked list of integers. b) Delete a given integer from the above doubly linked list. c) Display the contents of the above list after deletion
6. Write a program to implement Circular linked list operations
7. Determine the in degree and out degree of all the vertices of a given graph.
8. Write programs for implementing the following graph traversal algorithms:
 - a. Depth first traversal
 - b. Breadth first traversal
9. Determine whether the given graph is connected graph or not.
10. Write a program that uses functions to perform the following: a) Create a binary search tree of characters. b) Traverse the above Binary search tree recursively in Postorder.
11. Write a program that uses functions to perform the following: a) Create a binary search tree of integers. b) Traverse the above Binary search tree non-recursively in inorder.
12. Write a program to implement B-tree
13. Write programs for implementing the following sorting methods to arrange a list of integers in ascending order: a) Bubble sort b) Insertion sort
14. Write programs for implementing the following sorting methods to arrange a list of integers in ascending order: a) Quick sort b) Selection sort
15. Write programs for implementing the following sorting methods to arrange a list of integers in ascending order: a) Merge sort b) Heap sort
16. Write a program to search for a given element using
 - a. Linear search
 - b. Binary search

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L	H	L	L	-	-	-	L	-	-	-	-	M	H	L
CO2	-	L	H	-	M	-	-	L	-	-	-	-	M	H	L
CO3	-	L	-	H	H	-	-	L	-	-	-	-	M	H	L
CO4	-	L	H	L	-	-	-	L	-	-	-	-	M	H	L

20CST32: FUNDAMENTALS OF COMPUTER ORGANIZATION

Credits – 3
L:T:P :: 3:0:0

Sessional Marks: 30
University Exam Marks: 70

Course Objectives

To expose the students to the following:

1. Concepts of the basic structure and operation of the functional modules of a digital computer.
2. Analysis and design the functional modules, memory & I/O modules and pipelined processors in a digital computer.
3. Apply contextual knowledge to societal related issues.

Course Outcomes

After successful completion of course the student should be able to

- CO1** Demonstrate knowledge on Computer Arithmetic units, Register Transfer and Computer Instructions, Design of Control Unit, Input Output Organization and Memory system, Pipelining and Multiprocessing.
- CO2** Analyse the functional units of a digital computer.
- CO3** Design the functional modules in a digital computer - Arithmetic Units, Memory and I/O.
- CO4** Investigate the performance of memory, I/O, and pipelined processors.
- CO5** Select appropriate techniques of I/O, Pipelining and Multiprocessing to solve computing problems.
- CO6** Apply contextual knowledge of computer systems development to societal applications.

UNIT I

Register Transfer and Micro operations: Register transfer, Bus and memory transfers, Arithmetic micro-operations, Logic micro-operations, Shift micro operations, Arithmetic logic shift unit.

Computer Arithmetic: Fixed point representation, Floating point representation, Addition and subtraction, Binary multiplication algorithms, Binary division algorithms.

UNIT II

Basic Computer Organization and Design: Instruction codes, Computer registers, Computer instructions, Instruction formats, Addressing modes, Timing and control, Instruction cycle, Memory reference instructions, Input - Output and Interrupt.

Micro Programmed Control: Control memory, Address sequencing, Design of control unit, Hardwired control, Micro programmed control.

UNIT III

Input-Output Organization: Peripheral devices, Input-Output interface, Modes of transfer, Priority interrupt, Direct Memory Access, Input-Output Processor (IOP).

UNIT IV

The Memory System: Semiconductor RAM memories – Internal organization, Static memories, Synchronous and Asynchronous DRAMs, Structure of larger memories; Read- Only memories, Cache memories – Mapping functions; Secondary Storage – Magnetic Disks, Optical Disks.

UNIT V

Pipeline and Vector Processing: Parallel processing, Pipelining, Arithmetic pipeline, Instruction pipeline, Vector processing, Array processors.

Multiprocessors: Characteristics of multiprocessors, Interconnection structures, Inter-processor

arbitration, Inter-processor communication and synchronization.

Textbooks

1. Morris Mano, “Computer System Architecture”, Third Edition, Pearson Education, 2007.
2. Carl V. Hamacher, Zvonko G. Vranesic and Safwat G. Zaky, “Computer Organization”, Fifth Edition, McGraw-Hill, 2002.

Reference Books

1. William Stallings, —Computer Organization and Architecture: Designing for Performance, Seventh Edition, Pearson Education, 2007.
2. John P. Hayes, “Computer Architecture and Organization”, Third Edition, McGraw-Hill, 2002

Web References

1. <https://nptel.ac.in/courses/106/106/106106166/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	L	-	-	-	-	-	-	-	-	-	-	H	L	-
CO2	-	H	-	L	-	-	-	-	-	-	-	-	H	-	-
CO3	-	L	H	-	-	-	-	-	-	-	-	-	H	L	-
CO4	-	-	-	H	-	-	-	-	-	-	-	-	H	L	-
CO5	-	-	-	M	H	-	-	-	-	-	-	-	M	H	-
CO6	-	-	-	-	M	-	-	-	-	-	-	-	H	M	-

20CST33: BASICS OF COMPUTER NETWORKS

Credits – 3
L:T:P :: 3:0:0

Sessional Marks: 30
University Exam Marks: 70

Course Objectives

To expose the students to the following:

1. Modern network architectures from a design and performance perspective.
2. Major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
3. Network programming and WLAN metrics.

Course Outcomes

After successful completion of course the student should be able to

- CO1** Understand and explore the basics of Computer Networks and Various Protocols.
CO2 Administrate a network and schedule flow of information.
CO3 Examine the network security issues in Mobile and adhoc networks.
CO4 Demonstrate the TCP/IP and OSI fashions with merits and demerits.
CO5 Evaluate the shortest path by using Routing algorithms.
CO6 Design the various layer protocols.

UNIT I

Introduction: Introduction to computer networks, network hardware, Reference models. Overview of Physical layer

Data Link Layer: Design issues, error detection and correction, elementary data link protocols, sliding window protocols.

UNIT II

Medium Access Sub layer: Channel allocation problem, multiple access protocols, IEEE standard 802 for LANs and MANs, Bridges, High-speed LANs, Satellite network.

UNIT III

Network Layer: Design issues, routing algorithms, congestion control algorithms, internet working, the network layer in the internet, the network layer in ATM network.

UNIT IV

Transport Layer: Transport services, elements of transport protocols, a simple transport protocol, the internet transport protocols, (TCP and UDP), the ATN ALL layer protocols, performance issues.

UNIT V

Application Layer: Network security, DNS – Domain Name System, SNMP – Simple Network Management protocol, Electronic Mail, Usenet news, the World Wide Web (WWW), Multimedia.

Textbooks

1. Andrew S. Tanenbaum, David J. Wetherall, “Computer Networks”, 5th Edition, Pearson, 2011.

Reference Books

1. Behrouz A. Forouzan, “Data Communication and Networking”, 5th Edition, TMH, 2013.

Web References

1. <https://nptel.ac.in/courses/106/105/106105081/>
2. <https://nptel.ac.in/courses/106/106/106106091/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	-	-	-	-	L	-	-	-	-	-	H	-	-
CO2	-	H	H	-	-	-	M	-	-	-	-	-	H	-	-
CO3	-	M	-	H	-	M	M	-	-	-	-	-	H	-	M
CO4	-	H	-	-	-	-	-	-	-	-	M	-	H	-	-
CO5	L	M	-	H	-	-	-	-	-	-	-	-	L	M	-
CO6	-	-	H	M	-	-	-	-	L	L	-	-	L	M	H

20CST34: INTRODUCTION TO CYBER SECURITY**Credits – 3****L:T:P :: 3:0:0****Sessional Marks: 30****University Exam Marks: 70**

Course Objectives

To expose the students to the following

1. Instigate cyber threats and cyber security and to facilitate the awareness in the times of growing cyber-crime episodes.
2. Learning how cyber security is going to help to understand the implications of cybercrime.
3. Facilitating an idea about the legal perspectives and laws related to cybercrimes in Indian context.
4. Familiarize how to apply security and privacy methods in development of modern applications and in organizations to protect people and to prevent cybercrimes.

Course Outcomes

After successfully completion of course the student should be able to

- CO1** Analyze various aspects of Cyber security, Cyber-crimes and its related laws in Indian and Global act.
- CO2** Understand how cyber security is going to help the implications of cybercrime.
- CO3** Examine the legal perspectives and laws related to cybercrimes in Indian context.
- CO4** Apply security and privacy methods in development of modern applications and in organizations to protect people and to prevent cyber-crimes.

UNIT I

Introduction to Cyber Crimes: Introduction, Definition, Origin, Cyber Crime and Information Security, Cyber Criminals, Classifications of Cyber Crimes, The Legal Perspectives and Indian Perspective, Cyber Crime and Indian ITA 2000, Global Perspective on Cyber Crimes.

Cyber Offenses: Introduction, Criminals Planning on Attacks, Social Engineering, Cyber Stalking, Cyber Café and Crimes, Botnets.

UNIT II

Tools and Methods used in Cyber Crime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan horses and Backdoors, Steganography, DoS and DDoS attacks. Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).

UNIT III

Cyber Crimes and Cyber Security-Legal Perspectives: Introduction, Cyber Crime and the legal landscape around the world. Cyber Laws in Indian Context, The Indian IT Act, Challenges to Indian Law and Cyber Crime Scenario in India, Consequences of not addressing the weakness in IT Act, Digital Signatures and the Indian IT Act, Cyber Crime and Punishment, Cyberlaw, Technology and Students in India Scenario.

UNIT IV

Cyber Security-Organizational Implications: Introduction, Cost of Cyber Crimes and IPR issues, Web Threats for Organizations - Evils and Perils, Security and Privacy Implications from Cloud Computing, Social Media Marketing-Security Risks and Perils for Organizations.

UNIT V

Cyber Security-Organizational Implications: Social Computing and Associated Challenges for Organizations, Protecting People's Privacy in Organization, Organizational Guidelines for Internet Usage, Safe Computing and Usage Policy, Incident Handling and Best Practices, Media and Asset Protection.

Textbooks

1. Nina Gobole, Sunit Belapure, –Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, 1st Edition, Wiley India, 2011.

Reference Books

1. Robert Bird, Jonathan J. Darrow, Gerald R. Ferrera, Jacqueline Klosek, Margo E. K. Reder, Stephen Lichtenstein, Jeffrey Aresty, “Cyber Law: Text and Cases”, 3rd Edition, Cengage Learning, 2012.
2. Vivek Sood, “Cyber Law Simplified”, 1st Edition, Tata McGraw-Hill, 2012.
3. Prashant Mali, “Cyber Law and Cyber Crimes”, Snow White Publications Pvt. Ltd., 2013.

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	M	-	H	-	-	-	-	-	L	-	-	H	-	-
CO2	M	-	-	H	-	L	-	-	-	-	-	-	H	-	-
CO3	M	H	-	-	-	-	-	-	-	-	-	-	H	-	-
CO4	L	L	H	M	H	H	L	L	-	-	-	-	M	H	-

20CST35: INTRODUCTION TO CLOUD COMPUTING**Credits – 3**
L:T:P :: 3:0:0**Sessional Marks: 30**
University Exam Marks: 70**Course Objectives**

To expose the students to the following:

1. Fundamental concepts in the area of cloud computing.
2. Applications of cloud computing.
3. Cloud architecture and model.
4. The concept of virtualization and design of cloud services.
5. Illustrate the familiarity of the lead players in the cloud.
6. Evaluate the features of Cloud Simulator.

Course Outcomes

After successful completion of course the student should be able to

CO1. Define cloud computing and related concepts.

CO1 Know the key dimensions of the challenges and benefits of Cloud Computing.

CO2 Comprehend the hardware necessary for cloud computing and how components fit together.
CO4. Determine the suitability of in-house v/s hosted solutions.

CO3 Understand the systems, protocols and mechanisms to support cloud computing and develop applications for cloud computing.

CO4 Identify numerous opportunities exist for practitioners seeking to create solutions for cloud computing.

UNIT I

Systems Modelling, Clustering and Virtualization: Distributed System Models and Enabling Technologies. Computer Clusters for Scalable Parallel Computing. Virtual Machines and Virtualization of Clusters and Data centres.

UNIT II

Foundations: Introduction to Cloud Computing, Migrating into a Cloud, Enriching the Integration as a Service Paradigm for the Cloud Era. The Enterprise Cloud Computing Paradigm.

UNIT III

Infrastructure as a Service (IAAS) & Platform and Software as a Service (PAAS / SAAS): Virtual machines provisioning and Migration services, On the Management of Virtual machines for Cloud Infrastructures, Enhancing Cloud Computing Environments using a cluster as a Service. Secure Distributed Data Storage in Cloud Computing. Aneka, Comet Cloud, T-Systems, Understanding Scientific Applications for Cloud Environments

UNIT IV

Monitoring, Management and Applications: Architecture for Federated Cloud Computing, SLA Management in Cloud Computing, Performance Production for HPC on Clouds, Best Practices in Architecture Cloud Applications in the AWS cloud, Building Content Delivery networks Clouds

UNIT V

Governance and Case Studies: Organizational Readiness and Change management in the Cloud age. Data Security in the Cloud, Legal issues in Cloud computing. Achieving Production Readiness for Cloud

Services.

Textbooks

1. Rajkumar Buyya, “Cloud Computing: Principles and Paradigms”, John Wiley & Sons Inc., 2011.
2. Kal Hwang. Geoffey C. Fox. Jack J. Dongarra, “Distributed and Cloud Computing”, Elsevier, 2012

Reference Books

1. Anthony T. Velte. Toby J. VeFte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, Tata McGraw Hill, 2011.
2. Gautam Shroif, “Enterprise Cloud Computing”, Cambridge University Press, 2010.

Web References

1. <https://nptel.ac.in/courses/106/105/106105167/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	-	-	-	-	-	-	-	-	M	-	-	H	-	-
CO2	H	-	-	-	-	-	-	-	-	M	-	-	H	-	-
CO3	H	-	-	-	M	-	-	-	-	-	-	-	M	H	-
CO4	L	H	-	-	-	-	-	-	-	M	-	-	H	-	L
CO5	H	-	-	-	M	-	-	-	-	-	-	-	H	-	M
CO6	-	-	H	H	M	-	-	-	-	-	-	-	M	-	H

20CST36: INTRODUCTION TO ARTIFICIAL INTELLIGENCE**Credits–3**
L:T:P::3:0:0**Sessional Marks:30**
University Exam Marks:70**Course Objectives**

To expose the students to the following:

1. About various AI domains and problem solving techniques.
2. Basic proficiency in representing difficult real life problems in a state space representation so as to solve them uses AI techniques like searching.
3. The concept of Knowledge representations, its various approaches and issues, Non-monotonic environment and Symbolic Reasoning in Uncertainty.
4. Formal foundation on Strong & Weak slot & filler structures.

Course Outcomes

After successful completion of course the student should be able to

- CO1. Recognize various AI domains and identify problem solving techniques to apply them in real time applications.
- CO2. Analyze and formalize the problem as a state space, graph, design heuristics and select amongst different search based techniques to solve them.
- CO3. Represent Knowledge in propositional calculus and Predicate calculus.
- CO4. Demonstrate working knowledge of reasoning in the presence of incomplete and/or uncertain information.
- CO5. Get wide exposure about strong and weak slot & fillers available.

UNIT I

Introduction to AI: The AI Problems-The Underlying Assumption-What is an AI Technique-Tic-Tac-Toe game playing, Problems, Problem Spaces and Search-Defining the problem as a State Space Search- Production Systems-Control Strategies-Heuristic Search, Issues in the design of search program.

UNIT II

Heuristic search techniques: Generate and Test, Hill Climbing-Simple Hill Climbing-Steepest Ascent Hill Climbing-Simulated Annealing, Best-first-search-OR Graphs-A* Algorithm-Agenda Driven Search.

UNIT III

Knowledge representation: Knowledge Representation Issues- Representations in Mappings, Approaches to Knowledge representation, Issues in Knowledge Representation.

Predicate logic: Representing simple facts in Logic, representing instance and is a Relationships, Computable Function and Predicates, Resolution-Conversion to Clause form-The basics of Resolution-Resolution in Propositional Logic-Resolution in Predicate Logic, Natural deductions.

UNIT IV

Symbolic Reasoning under Uncertainty: Introduction to non-monotonic reasoning, logics for non-monotonic reasoning-Default Reasoning-Minimalist Reasoning, Implementation issues, Implementation in depth first search-Dependency directed Backtracking-Justification Based Truth Maintenance Systems-Logic-Based Truth Maintenance Systems- Implementation in Breadth first search.

UNIT V

Weak slot and Filler Structures: Semantic Nets- Intersection Search-Representing Non-binary Predicates-Partitioned Semantic Nets, Frames-Frames as Sets and Instances-Slots as Full-Fledged Objects-Slot-Values as Objects-Inheritance Revisited.

Textbooks

1. Elaine Rich, Kelvin Knight and Shiva Shankar B.Nair, –Artificial Intelligence, 3rd Edition, Tata McGrawHill, July, 2017.

Reference Books

1. Saroj koushik, —Artificial Intelligence, 1st Edition, Engage learning,2011
2. Elakumar, –Artificial Intelligence, 1st Edition, I.K.International publishing house,2010.

Web References

- 1.<https://nptel.ac.in/courses/106/105/106105077/>
- 2.<https://nptel.ac.in/courses/106/105/106105079/>
- 3.<https://nptel.ac.in/courses/106/106/106106140/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	-	-	-	-	L	-	-	-	-	-	H	-	L
CO2	-	H	-	M	-	-	-	L	-	-	-	-	H	L	L
CO3	H	-	-	-	-	-	-	-	-	-	-	-	H	-	-
CO4	-	H	-	M	-	-	-	-	-	-	-	-	H	-	-
CO5	H	-	M	-	-	-	-	L	-	-	-	-	H	L	-

20CST37: INTRODUCTION TO JAVA PROGRAMMING**Credits–3**
L:T:P::3:0:0**Sessional Marks:30**
University Exam Marks:70

Course Objectives

To expose the students to the following:

1. Basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methodsetc.
2. The fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc and exception handlingmechanisms.
3. The principles of inheritance, packages and interfaces.

Course Outcomes

After successful completion of course the student should be able to

- CO1. Identify classes, objects, members of a class and relationships among them needed for a specific problem.
- CO2. Write Java application programs using OOP principles and proper program structuring.
- CO3. Demonstrate the concepts of polymorphism and inheritance.
- CO4. Compose Java programs to implement error handling techniques using exception handling and understand the importance of multi-threading.
- CO5. Build the internet-based dynamic applications using the concept of applets.
- CO6. Understand the process of graphical user interface design and implementation using AWT

UNIT I

The History and Evolution of Java: Java's Lineage, The Creation of java, how java changed the internet, Java's magic: The byte code, Servlets: java on the server side, java Buzzwords, Evolution of java.

An Overview of Java: Object Oriented Programming, Two control statements, Using blocks of codes, Lexical issues, and the java class Libraries.

Data Types, Arrays and Variables: Primitive Types, Integers, Floating-point Types, Characters, Booleans, literals, variables, Type conversion and casting, Automatic Type Promotion in Expressions, Arrays, strings, Pointers.

UNIT II

Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logic operators, The assignment operator, The ? Operator, Operator Precedence, Using Parentheses.

Control Statements: Java's selection Statements, Iteration statements, Jump Statements.

Introducing Classes: Class Fundamentals, Declaring Objects, Assuming Object reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The Finalize() method, A Stack class. Overloading Methods, Using Object as Parameter, Argument Passing, Returning Objects, Recursion, Introducing Access control, Understanding static, Introducing Nested and Inner classes, Exploring the String class, Using Command line Arguments, Varargs: variable-Length Arguments.

UNIT III

Inheritance: Basics, Using super, creating a multi-level hierarchy, when constructors are executed, method overriding, dynamic method dispatch, using abstract class, using final with inheritance, the object class.

Packages and Interfaces: Packages, Access protection, Importing Packages, Interfaces, Default Interfaces, Default interface methods, Use static methods in an Interface, Final thoughts on Packages and interfaces.

Exception Handling: Exception handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch clauses, Nested try statements, throw, throws, finally, Java Built-in Exceptions, Creating your own exception subclasses, Chained Exceptions, Three Recently added Exceptions features, Using Exceptions.

UNIT IV

Multithreaded Programming: The java Thread Model, The main thread , Creating Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, resuming and stopping threads, Obtaining a thread state, Using Multithreading.

I/O, Applets, and Other Topic: I/O basics, Reading Console input, Writing console Output, The PrintWriter class, Reading and writing files, Automatically closing a file, Applet fundamentals, enumerations type wrappers auto boxing annotations, Generics: The general form of a generics class, creating a generic method, generics interfaces.

UNIT V

Introduction the AWT: Working with windows, graphics and Text: AWT classes, window fundamentals, working with frame windows, creating a frame window in a an AWT Based applet, creating a window program, displaying information within a window, Graphics, working with color, setting the paint mode, working with fonts, managing text output using font metrics,.

Using AWT controls, Layout Mangers, and Menus: AWT control fundamentals, Labels, using buttons, applying check boxes, check box group, choice controls, using lists, Managing scroll bars, using a Text field, Using a Text area, understanding layout managers, Menu bars and Menus, dialog boxes, file dialog, Overriding paint().

Text Books

1. Herbert Schildt, —Java, The Complete Reference, 9th Edition, McGraw Hill Education, 2016.

Reference Books

1. B. Eswara Reddy, P. Raghavan, T. V. Suresh Kumar, –Programming with Javall, Pearson Edition, 2011
2. Herbert Schildt and Dale Skrien, –Java Fundamentals - A Comprehensive Introduction, Special Indian Edition, McGrawHill, 2013.

Web References

1. <https://nptel.ac.in/courses/106/105/106105191/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	-	-	-	-	-	-	-	-	-	-	M	H	-
CO2	L	M	H	-	-	-	-	-	-	-	-	-	M	H	-
CO3	-	M	H	L	-	-	-	-	-	-	-	-	M	H	-
CO4	-	H	M	-	-	-	-	-	-	-	-	-	M	H	-
CO5	L	-	H	-	M	-	-	-	-	-	-	-	M	H	-
CO6	M	L	H	-	-	-	-	-	-	-	-	-	H	M	-

20CST38: FUNDAMENTALS OF INTERNET OF THINGS**Credits–3**
L:T:P::3:0:0**Sessional Marks:30**
University Exam Marks:70

Course Objectives

To expose the students to the following:

1. Concepts IoT and python.
2. IoT Market perspective.
3. IoT Architecture, Real World IoT Design Constraints, Industrial Automation and Commercial Building Automation in IoT.

Course Outcomes

After successful completion of course the students should be able to

- CO1. Analyse the vision of IoT from a global context.
CO2. Understand the vision of IoT from a global context.
CO3. Determine the Market perspective of IoT.
CO4. Use devices like Raspberry PI-Interfaces, Gateways and Data Management in IoT.
CO5. Build architecture for IoT.

UNIT I

Introduction to Internet of Things: Definition and Characteristics of IoT, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs. IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels.

UNIT II

IoT and M2M: Software defined networks, network function virtualization, difference between SDN and NFV for IoT, Basics of IoT System Management with NETCOZF, YANG- NETCONF, YANG, and SNMP NETOPEER.

UNIT III

Introduction to Python: Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, Python packages - JSON, XML, HTTPLib, URLLib, SMTPLib.

UNIT IV

IoT Physical Devices and Endpoints: Introduction to Raspberry PI-Interfaces (serial, SPI, I2C) Programming – Python program with Raspberry PI with focus of interfacing external gadgets, controlling output, reading input from pins.

UNIT V

IoT Physical Servers and Cloud Offerings: Introduction to Cloud Storage models and Communication APIs, Web server for IoT, Cloud for IoT

Text Books

1. ArshdeepBahga and Vijay Madiseti, –Internet of Things - A Hands-on Approach, Orient Blackswan Private Limited - New Delhi, University Press, 2015.
2. Matt Richardson and Shawn Wallace, –Getting Started with Raspberry Pi, O'Reilly, Maker Media Inc,2013.

Reference Books

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, –From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligencell, 1st Edition, Academic Press,2014.
2. Michael Miller, –The Internet of Thingsll, First Edition, Pearson,2015.
3. Daniel Minoli, –Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communicationsll, Wiley,2013.

Web References

1. <https://nptel.ac.in/courses/106/105/106105166/>

Course Outcomes – Program Outcomes – Program Specific Outcomes (CO-PO-PSO) Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	H	-	M	-	-	-	-	-	-	-	-	H	-	-
CO2	H	-	-	-	M	-	L	-	-	-	-	-	H	-	-
CO3	M	-	-	-	-	-	H	-	-	M	-	L	H	-	-
CO4	-	-	-	-	H	-	-	-	-	-	-	-	L	H	M
CO5	-	M	H	-	-	-	-	-	-	-	-	-	L	-	H