

BCA (HONS) 5th SEMESTER
DISCIPLINE SPECIFIC COURSE (CORE)

OPTION - I

BCA520D1B: INFORMATION SECURITY

CREDITS: THEORY: 4; PRACTICAL: 2
MAX. MARKS: THEORY: 60; PRACTICAL: 30
MIN. MARKS: THEORY: 24; PRACTICAL: 12

THEORY: 60 LECTURES

UNIT I

1. Introduction (5 Lectures)

Security, Attacks, Computer Criminals, Security Services, Security Mechanisms.

2. Cryptography (10 Lectures)

Substitution ciphers, Transpositions Cipher, Confusion, diffusion, Symmetric, Asymmetric Encryption. DES Modes of DES, Uses of Encryption, Hash function, key exchange, Digital Signatures, Digital Certificates.

UNIT II (8 Lectures)

3. Program Security

Secure programs, Non malicious Program errors, Malicious codes virus, Trap doors, Salami attacks, Covert channels, Control against program

4. Threats. (7 Lectures)

Protection in OS: Memory and Address Protection, Access control, File Protection, User Authentication.

UNIT III (7 Lectures)

5. Database Security

Requirements, Reliability, Integrity, Sensitive data, Inference, Multilevel Security.

6. Security in Networks (8 Lectures)

Threats in Networks, Security Controls, firewalls, Intrusion detection systems, Secure e-mails

UNIT IV

7. Administrating Security (15 Lectures)

Security Planning, Risk Analysis, Organisational Security Policy, Physical Security. Ethical issues in Security: Protecting Programs and data. Information and law.

Recommended Books:

1. C. P. Pfleeger, S. L. Pfleeger; Security in Computing, Prentice Hall of India, 2006
2. W. Stallings; Network Security Essentials: Applications and Standards, 4/E, 2010

LAB: BCA520D1B: INFORMATION SECURITY (CREDITS: 2; LECTURES:60)

1. Demonstrate the use of Network tools: ping, ipconfig, ifconfig, tracert, arp, netstat, whois
2. Use of Password cracking tools: John the Ripper, Ophcrack. Verify the strength of passwords using these tools.
3. Perform encryption and decryption of Caesar cipher. Write a script for performing these operations.
4. Perform encryption and decryption of a Rail fence cipher. Write a script for performing these operations.
5. Use nmap/zenmap to analyse a remote machine.
6. Use Burp proxy to capture and modify the message.
7. Demonstrate sending of a protected word document.
8. Demonstrate sending of a digitally signed document.
9. Demonstrate sending of a protected worksheet.
10. Demonstrate use of steganography tools.
11. Demonstrate use of gpg utility for signing and encrypting purposes.

BCA (HONS) 5th SEMESTER
DISCIPLINE SPECIFIC COURSE (CORE)
OPTION - II

BCA521C2: PROGRAMMING IN JAVA

CREDITS: THEORY: 4; TUTORIAL: 2
MAX. MARKS: THEORY: 60; TUTORIAL: 30
MIN. MARKS: THEORY: 24; TUTORIAL: 12

THEORY: 60 LECTURES

UNIT-I

1. Introduction to Java (6 Lectures)

Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

2. Arrays and Strings (9 Lectures)

Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects,

Manipulating Strings, String Immutability & Equality, Passing Strings To& From Methods, String Buffer Classes.

UNIT-II

3. Object-Oriented Programming (10 Lectures)

Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

4. I/O (5 Lectures)

Simple I/O using System. Out and the Scanner class, Byte and Character streams,

Reading/Writing from console and files.

UNIT-III

5. Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata (15 lectures)

Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

UNIT-IV

6. Exception Handling, Threading, Networking and Database Connectivity (15 Lectures)

Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

Reference Books

1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.
2. James Gosling, Bill Joy, Guy L Steele Jr, Gilad Bracha, Alex Buckley "The Java Language Specification, Java SE 8 Edition (Java Series)", Published by Addison Wesley, 2014.
3. Joshua Bloch, "Effective Java" 2nd Edition, Publisher: Addison-Wesley, 2008.
4. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 1, 9th Edition, Printice Hall. 2012
5. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features", 9th Edition, Printice Hall. 2013
6. Bruce Eckel, "Thinking in Java", 3rd Edition, PHI, 2002.
7. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill. 2009.
8. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.
9. "Head First Java", Orielly Media Inc. 2nd Edition, 2005.
10. David J. Eck, "Introduction to Programming Using Java", Published by CreateSpace Independent Publishing Platform, 2009.
11. John R. Hubbard, "Programming with JAVA", Schaum's Series, 2nd Edition, 2004.

LAB: PROGRAMMING IN JAVA (2 CREDITS)

1. To find the sum of any number of integers entered as command line arguments
 2. To find the factorial of a given number
 3. To learn use of single dimensional array by defining the array dynamically.
 4. To learn use of. lenth in case of a two dimensional array
 5. To convert a decimal to binary number
 6. To check if a number is prime or not, by taking the number as input from the keyboard
 6. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
 7. Write a program that show working of different functions of String and StringBufferclasss like setCharAt(),setLength(), appendO, insertO, concat()and equals().
 8. Write a program to create a —distance!! class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
 9. Modify the —distancel class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
 10. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)
 11. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword
 12. Write a program to show the use of static functions and to pass variable length arguments in a function.
 13. Write a program to demonstrate the concept of boxing and unboxing.
 14. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
 15. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacii series is given in a different file belonging to the same package.
 16. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
 17. Write a program —DivideByZero that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
 18. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
 19. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Trowable).
 20. Write a program to demonstrate priorities among multiple threads.
 21. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
 22. Write a program to create URL object, create a URLConnection using the openConnection() method and then use it examine the different components of the URLand content.
 23. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
 24. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet's window.
 25. Write a program to get the URL/location of code (i.e. java code) and document (i.e. html file).
 26. Write a program to demonstrate different mouse handling events like mouseClicked(), mouseEnteredO, mouseExitedO, mousePressed, mouseReleased() and mouseDragged().
 27. Write a program to demonstrate different keyboard handling events.
- Write a program to generate a window without an applet window using mainO function. 30. Write a program to demonstrate the use of push buttons.