



# INSTITUTE OF AERONAUTICAL ENGINEERING

Dundigal, Hyderabad -500 043

## COMPUTER SCIENCE AND ENGINEERING

### TUTORIAL QUESTION BANK

<b>Course Name</b>	<b>JAVA PROGRAMMING</b>
<b>Course Code</b>	A40503
<b>Class</b>	II B. Tech II Semester
<b>Branch</b>	Computer Science and Engineering
<b>Year</b>	2016 – 2017
<b>Course Faculty</b>	Mr. P. Ravinder , Assistant Professor, CSE Mr. N V Krishna Rao, Assistant Professor, CSE Ms. K. Radhika Assistant Professor, CSE

### OBJECTIVES

To meet the challenge of ensuring excellence in engineering education, the issue of quality needs to be addressed, debated and taken forward in a systematic manner. Accreditation is the principal means of quality assurance in higher education. The major emphasis of accreditation process is to measure the outcomes of the program that is being accredited.

In line with this, Faculty of Institute of Aeronautical Engineering, Hyderabad has taken a lead in incorporating philosophy of outcome based education in the process of problem solving and career development. So, all students of the institute should understand the depth and approach of course to be taught through this question bank, which will enhance learner's learning process.

S. No	Question	Blooms Taxonomy Level	Course Outcom e
<b>UNIT - I</b>			
<b>PART – A (Short Answer Questions)</b>			
1	<b>Define</b> OOP?	Remembering	1,2,3
2	<b>Distinguish</b> between procedural language and OOPs?	Analyzing	1,2,3
3	<b>Define</b> Encapsulation?	Remembering	1,2,3
4	<b>Define</b> Inheritance?	Remembering	1,2,3
5	<b>Define</b> Polymorphism?	Remembering	1,2,3
6	<b>List</b> advantages of OOP?	Remembering	1,2,3
7	<b>List</b> disadvantages of OOP?	Remembering	1,2,3
8	<b>Explain</b> briefly out history of java?	Understanding	1,2,3
9	<b>Explain</b> briefly about different types of data types in java?	Understanding	1,2,3
10	<b>What</b> is constant? Explain different types of constants?	Remembering	1,2,3
11	<b>What</b> is Variable? <b>Describe</b> scope and life time of variables?	Remembering	1,2,3
12	<b>List</b> and describe different types of operators?	Remembering	1,2,3
13	<b>Define</b> type conversion?	Remembering	1,2,3
14	<b>Define</b> type casting?	Remembering	1,2,3
15	<b>Define</b> enumerated types?	Remembering	1,2,3
16	<b>Define</b> an array?	Remembering	1,2,3
17	<b>Define</b> this reference?	Remembering	1,2,3

18	<b>Define</b> constructor?	Remembering	1,2,3
19	<b>Define</b> recursion?	Remembering	1,2,3
20	<b>Define</b> garbage collection?	Remembering	1,2,3
<b>Part - B (Long Answer Questions)</b>			
1	<b>Discuss</b> the various characteristics of object oriented programming concepts?	Creating	1,2,3
2	<b>Explain</b> briefly about the features (buzzwords) of Java.	Understanding	1,2,3
3	<b>Discuss</b> various Differences between Java and C++.	Creating	1,2,3
4	<b>Why</b> Java is a pure object oriented programming language	Remembering	1,2,3
5	<b>Distinguish</b> between applications and applets in Java?	Analyzing	1,2,3
6	<b>Explain</b> the importance of this keyword with an example.	Understanding	1,2,3
7	<b>What</b> is method overloading? Explain with an example.	Remembering	1,2,3
8	<b>Discuss</b> about the constructor overloading with an example.	Creating	1,2,3
9	<b>What</b> is Array? Explain the concept of arrays with an example.	Remembering	1,2,3
10	<b>Explain</b> briefly about String class and discuss various methods in string class with an example.	Understanding	1,2,3
11	<b>Explain</b> about the console input and output with an example.	Understanding	1,2,3
12	<b>Discuss</b> about various conditional statements with necessary examples	Creating	1,2,3
13	<b>Explain</b> about different loops with an example.	Understanding	1,2,3
14	<b>What</b> is the use of break and continue statements in java? Explain with an example.	Remembering	1,2,3
15	<b>Discuss</b> about the operator hierarchy with an example.	Creating	1,2,3
16	<b>What</b> is the use of the operators in java? Explain with an example.	Remembering	1,2,3
17	<b>Define</b> static field? Write with an example.	Remembering	1,2,3
18	<b>Define</b> static method? Write with an example.	Remembering	1,2,3
19	<b>What</b> is type conversion and casting? Explain with an example.	Remembering	1,2,3
20	<b>Explain</b> about foreach loop with an example	Understanding	1,2,3
<b>Part - C (Problem Solving and Critical Thinking Questions)</b>			
1	Student john12 = new Student(1001, "John", 12); Student john13 = new Student(1002, "John", 13); <b>System.out.println</b> ("comparing John, 12 and John, 13 with compareTo : " + john12.compareTo(john13)); <b>then predict</b> the output of the code?	Understand	1,2,3
2	<b>What</b> is the output of the program? class Lifetime { public static void main(String args[]) { int x; for (x=0; x<3; x++) { int y=-1; System.out.println(" y is : " + y); y=100; System.out.println(" y is now : " + y); } } }	Understand	1,2,3
3	<b>What</b> will be the output of the program? public class If2 { static boolean b1, b2; public static void main(String [] args) { int x = 0; if ( !b1 )	Understand	1,2,3

	<pre> {     if ( !b2 )     {         b1 = true;         x++;         if ( 5 &gt; 6 )         {             x++;         }         if ( !b1 )             x = x + 10;         else if ( b2 = true )             x = x + 100;         else if ( b1   b2 )             x = x + 1000;         }     }     System.out.println(x); } </pre>		
4	<p><b>Explain</b> the following code legal in Java? is it example of method overloading or overriding?</p> <pre> public String getDescription(Object obj){     return obj.toString; } public String getDescription(String obj){     return obj; } and public void getDescription(String obj){     return obj; } </pre>	Understand	1,2,3
5	<p><b>Analyze</b> the following program and find the output of the program?</p> <pre> public class CounterAtomic {     private AtomicLong counter = new AtomicLong();     public void increment() {         counter.incrementAndGet();     }     public long get() {         return counter.get();     } } </pre>	Analyze	1,2,3
6	<p><b>What</b> will be the output of the program?</p> <pre> public class Test {     public int aMethod()     {         static int i = 0;         i++;         return i;     }     public static void main(String args[])     {         Test test = new Test();         test.aMethod();         int j = test.aMethod();     } } </pre>	Remembering	1,2,3

	<pre> System.out.println(j);     } } </pre>		
7	<p><b>What</b> will be the output of the program?</p> <pre> public class Test {     public static void main(String args[])     {         int i = 1, j = 0;         switch(i)         {             case 2: j += 6;             case 4: j += 1;             default: j += 2;             case 0: j += 4;         }         System.out.println("j = " + j);     } } </pre>	Remembering	1,2,3
8	<p><b>Analyze</b> the following program and find the output of the program?</p> <pre> Class Test {     public static void main(String args[])     {         int x, y;         y=20;         for(x=0; x&lt;10: x++)         {             System.out.println("this is x:"+x);             System.out.println("this is y:" +y);             y= y-2;         }     } } </pre>	Analyze	1,2,3
9	<p><b>What</b> will be the output of the program?</p> <pre> class BitShift {     public static void main(String [] args)     {         int x = 0x80000000;         System.out.print(x + " and ");         x = x &gt;&gt;&gt; 31;         System.out.println(x);     } } </pre>	Remembering	1,2,3
10	<p><b>Analyze</b> and find out What will be the problem in the program?</p> <pre> class Equals {     public static void main(String [] args)     {         int x = 100;         double y = 100.1;         boolean b = (x = y);         System.out.println(b);     } } </pre>	Analyze	1,2,3

**UNIT - II**

**Part – A (Short Answer Questions)**

1	<b>Define</b> Inheritance?	Understand	4
2	<b>List</b> types of inheritances in java?	Remembering	4
3	<b>What</b> are Member access rules in java?	Remembering	4
4	<b>Explain</b> the Uses of „Super“ keyword	Understanding	4
5	<b>Explain</b> the Uses „final“ keyword with inheritance	Understanding	4
6	<b>What</b> is Object class?	Understand	4
7	<b>Define</b> abstract classes?	Remembering	4
8	<b>Define</b> polymorphism?	Remembering	4
9	<b>Define</b> dynamic binding?	Remembering	4
10	<b>Define</b> method overriding?	Remembering	4
11	<b>Compare and Contrast</b> differences between interfaces vs. Abstract classes	Understanding	4
12	<b>Define</b> interface?	Remembering	4
13	<b>Define</b> inner classes?	Remembering	4
14	<b>Define</b> static inner classes?	Remembering	4
15	<b>Define</b> a package?	Remembering	4
16	<b>Define</b> various steps for creating and importing packages.	Understand	4
17	<b>Define</b> abstract methods?	Remembering	4
18	<b>Define</b> CLASSPATH?	Remembering	4
19	<b>List</b> advantages of inheritance?	Remembering	4
20	<b>How</b> we implement an interface write steps	Remembering	4

**Part - B (Long Answer Questions)**

1	<b>Define</b> Inheritance? Discuss its uses and Hierarchical abstractions?	Remembering	4
2	<b>List</b> different types of inheritances in java? Explain each of them in detail with an example programs.	Remembering	4
3	<b>Discuss</b> about Object class in detail	Creating	4
4	<b>Explain</b> the Uses of „Super“ keyword , discuss accessing the member of a super class.	Understanding	4
5	<b>What</b> is package? Discuss its advantages?	Remembering	4
6	<b>Explain</b> different Types of Packages	Understanding	4
7	<b>Define</b> dynamic binding? Write with an example?	Remembering	4
8	<b>Define</b> method overriding? Write with an example?	Remembering	4
9	<b>Define</b> Abstract classes? Write with an example?	Remembering	4
10	<b>Define</b> interface? Write with an example?	Remembering	4
11	<b>Define</b> inner classes? Write with an example?	Remembering	4
12	<b>Discuss</b> in detail about creating and importing packages with an examples	Creating	4
13	<b>Discuss</b> in detail how packages are accessed	Creating	4
14	<b>Explain</b> different ways to extending interfaces with an example	Understanding	4
15	<b>Define</b> interface? Write Differences between classes and interfaces?	Remembering	4
16	<b>What</b> is final keyword? Explain its importance in java with an example program.	Understand	4
17	<b>What</b> is inheritance? Explain the benefits of inheritance with an example	Remembering	4
18	<b>What</b> are various Member access rules explain with an example	Remembering	4
19	<b>Discuss</b> the various levels of Access protection available for packages and their implications.	Creating	4
20	<b>Compare</b> and contrast overloading and overriding methods	Remembering	4

**Part – C (Problem Solving and Critical Thinking)**

1	<p><b>Analyze</b> the program and give output</p> <pre> public class Foo {     Foo()     {         System.out.print("foo");     } } class Bar {     Bar()     {         System.out.print("bar");     } } public void go() {     System.out.print("hi"); } } /* class Bar ends */ public static void main (String [] args) {     Foo f = new Foo();     f.makeBar(); } void makeBar() {     (new Bar() {}).go(); } } /* class Foo ends */ </pre>	Analyzing	4
2	<p><b>Explain</b> the following code legal in Java?</p> <pre> class OuterClass {     private int privInt = 10;     public void createInnerClass() {         InnerClass inClass = new InnerClass();         inClass.accessOuter();     } } class InnerClass {     public void accessOuter() {         System.out.println("The outer class's privInt is " + privInt);     } } } public static void main(String[] args) {     OuterClass outClass = new OuterClass();     OuterClass.InnerClass inner = outClass.new InnerClass();      inner.accessOuter(); } </pre>	Understanding	4
3	<p><b>Analyze</b> the following program and find the output of the program?</p> <pre> class A {     //Members and methods declarations. } class B extends A {     //Members and methods from A are     inherited.     //Members and methods declarations of B. } </pre>	Analyze	4

	<pre> } class A { public A() {      System.out.println("New A"); } } class B extends A { public B() { super(); System.out.println("New B"); } } </pre>		
4	<p><b>Illustrate</b> the output of the following program?</p> <pre> interface MyInterface { public void method1(); public void method2(); } class XYZ implements MyInterface { public void method1() { System.out.println("implementation of method1"); } public void method2() { System.out.println("implementation of method2"); } public static void main(String arg[]) { MyInterface obj = new XYZ(); obj. method1(); } } </pre>	Applying	4
5	<p><b>What</b> is the output of the program?</p> <pre> Class A { final public int GetResult(int a, int b) { return 0; } } class B extends A { public int GetResult(int a, int b) {return 1; } } public class Test { public static void main(String args[]) { B b = new B(); System.out.println("x = " + b.GetResult(0, 1)); } } </pre>	Remembering	4
6	<p><b>What</b> will be the output of the program?</p> <pre> class Super { public int i = 0; public Super(String text) { i = 1; } } </pre>	Remembering	4

	<pre> } class Sub extends Super { public Sub(String text) { i = 2; } public static void main(String args[]) { Sub sub = new Sub("Hello"); System.out.println(sub.i); } } </pre>		
7	<p><b>What</b> will be the output of the program?</p> <pre> interface Count { short counter = 0; void countUp(); } public class TestCount implements Count { public static void main(String [] args) { TestCount t = new TestCount(); t.countUp(); } public void countUp() { for (int x = 6; x&gt;counter; x--, ++counter) { System.out.print(" " + counter); } } } </pre>	Remembering	4
8	<p><b>Analyze</b> and find out the output of the program?</p> <pre> public class Test { public int aMethod() { static int i = 0; i++; return i; }  public static void main(String args[]) { Test test = new Test(); test.aMethod(); int j = test.aMethod(); System.out.println(j); } } </pre>	Analyze	4
9	<p><b>Develop</b> a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area().provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method</p>	Applying	4



	print Area () that prints the area of the given shape.		
10	<p><b>Analyze</b> and find out the output of the program?</p> <pre> package mypack class Book  { String bookname; String author; Book(String b, String c) { this.bookname = b; this.author = c; } public void show() { System.out.println(bookname+" "+ author); } } class test { public static void main(String[] args) { Book bk = new Book("java","Herbert"); bk.show(); }} </pre>	Analyze	4

### UNIT-III

#### Part - A (Short Answer Questions)

1	<b>Define</b> Exception?	Remembering	1
2	<b>Distinguish</b> between exception and error?	Analyzing	2
3	<b>What</b> are the benefits of exception handling	Remembering	3
4	<b>Explain</b> the classification of exceptions	Understanding	4
5	<b>Define</b> checked exceptions?	Remembering	5
6	<b>Define</b> unchecked exceptions?	Remembering	6
7	<b>Define</b> built in exceptions?	Remembering	7
8	<b>Explain</b> the usage of try and catch	Understanding	8
9	<b>Explain</b> the usage of throw, throws and finally	Understanding	9
10	<b>Distinguish</b> between throw and throws?	Analyzing	10
11	<b>Distinguish</b> between process and thread?	Analyzing	5,6
12	<b>What</b> are thread states? Explain.	Remembering	5,6
13	<b>What</b> are the different ways to create a thread?	Remembering	5,6
14	<b>Define</b> producer consumer problem?	Remembering	5,6
15	<b>Define</b> inter-thread communication?	Remembering	5,6
16	<b>How</b> threads are synchronized?	Remembering	5,6
17	<b>What</b> are different thread priorities	Remembering	5,6
18	<b>How</b> many ways can thread be Created?	Understanding	5,6
19	<b>Explain</b> about the alive() and join() method	Understanding	5,6
20	<b>Explain</b> about “thread class implements Runnable interface”	Understanding	5,6

#### Part – B (Long Answer Questions)

1	<b>Explain</b> briefly about exception handling mechanisms with example programs.	Understanding	5,6
---	---	---------------	-----

2	<b>What</b> are try, catch , and finally keywords in with an example	Remembering	5,6
3	<b>Define</b> throw keyword? Write with an example	Remembering	5,6
4	<b>Define</b> throws keyword? Write with an example	Remembering	5,6
5	<b>Define</b> a exception called “NotEqualException” that is thrown when a float value is not equal to 3.14. write a program that uses the above user defined exception.	Remembering	5,6
6	<b>Differentiate</b> between checked and unchecked exceptions?	Remembering	5,6
7	<b>Define</b> a exception ?explain the different type of exception.		
8	<b>Develop</b> a program to implement built in exceptions?	Apply, Create	5,6
9	<b>Explain</b> the various ways of creation of thread with an examples.	Understanding	5,6
10	<b>Explain</b> with an example how java performs thread synchronization?	Understanding	5,6
11	<b>Explain</b> the producer consumer problem with an example	Understanding	5,6
13	<b>Explain</b> briefly about the life cycle of a thread with an example.	Understanding	5,6
15	<b>Differentiate</b> between multiprocessing and multithreading?what is to be done to implement these in a pogram	Remembering	5,6
16	<b>Define</b> thread.How do we set priorities for threads?	Understanding	5,6
17	<b>Explain</b> Deamon threads with a an example?	Understanding	5,6
18	<b>Explain</b> any five methods of thread class?	Understanding	5,6
19	<b>Explain</b> with an example how thread class methods can be used to control the behavior of a thread?	Understanding	5,6
<b>Part – C (Problem Solving and Critical Thinking Questions)</b>			
1	<b>What</b> will be the output of the program? <pre> public class X { public static void main(String [] args) { try { badMethod(); System.out.print("A"); } catch (Exception ex)  { System.out.print("B"); } finally { System.out.print("C"); } System.out.print("D"); } public static void badMethod() { throw new Error(); } } </pre>	Remembering	5,6
2	<b>Analyze</b> the program and find the output ?	Analyze	5,6

	<pre> public class Test {  public static void aMethod() throws Exception  {  try    { throw new Exception();  }  finally { System.out.print("finally "); }  }  public static void main(String args[]) {  try  {  aMethod();  }  catch (Exception e) { System.out.print("exception "); }  System.out.print("finished");  }  } </pre>		
3	<p><b>What</b> will be the output of the program?</p> <pre> class s1 implements Runnable { int x = 0, y = 0; int  addX() {x++; return x;}  int  addY()  {y++; return y;}      public void run() { for(int i = 0; i &lt; 10; i++) System.out.println(addX() + " " + addY()); } public static void main(String args[]) { s1 run2 = new s1(); Thread t1      =      new Thread(run1);  Thread t2 = new Thread(run2); t1.start(); t2.start(); } } </pre>	Remembering	5,6
4	<p><b>Explain</b> the output of the following program?</p> <pre> class Exceptions { public static void main(String[] args) { String languages[] = { "C", "C++", "Java", "Perl", "Python" }; </pre>	Evaluated	5,6

	<pre> try { for (int c = 1; c &lt;= 5; c++) { System.out.println(languages[c]); } } catch (Exception e) { System.out.println(e); } } } </pre>		
5	<p><b>What</b> is the output of the below program?</p> <pre> class Allocate { public static void main(String[] args) {  try { long data[] = new long[1000000000];  }  catch (Exception e) { System.out.println(e);  }  finally { System.out.println("finally block will execute always.");  }  }  }then predict the output? </pre>	Remembering	5,6
6	<p><b>What</b> will be the output of the program?</p> <pre> class MyThread extends Thread {  public static void main(String [] args) {  MyThread t = new MyThread(); Thread x = new Thread(t); x.start();  }  public void run() { for(int i = 0; i &lt; 3; ++i) { System.out.print(i + ".."); } } } </pre>	Analyzing	5,6
7	<p><b>What</b> will be the output of the program?</p> <pre> public class RTEexcept { public static void throwit () { </pre>	Remembering	5,6

	<pre> System.out.print("throwit "); throw new RuntimeException(); } public static void main(String [] args) { try { System.out.print("hello "); throwit(); } catch (Exception re ) { System.out.print("caught "); } finally { System.out.print("finally "); } System.out.println("after "); } } </pre>		
8	<p><b>Analyze</b> the program and find the output</p> <pre> public class NFE { public static void main(String [] args) { String s = "42"; try { s = s.concat(".5"); double d = Double.parseDouble(s); s = Double.toString(d); int x = (int) Math.ceil(Double.valueOf(s).doubleValue());  System.out.println(x); } catch (NumberFormatException e) { System.out.println("bad number"); } }} </pre>	Analyzing	5,6
9	<p><b>What</b> will be the output of the program?</p> <pre> class MyThread extends Thread { MyThread() { System.out.print(" MyThread"); } public void run() { System.out.print(" bar"); } public void run(String s) { System.out.println(" baz"); } } </pre>	Remembering	5,6

	<pre> } public class TestThreads { public static void main (String [] args) { Thread t = new MyThread() { public void run() { System.out.println(" foo"); } } t.start(); } } </pre>		
--	---	--	--

10	<p><b>What</b> will be the output of the program?  class implements Runnable  {  int x, y;  public void run()  {  for(int i = 0; i &lt; 1000; i++)  synchronized(this)  {  x = 12;  y = 12;  }  System.out.print(x + " " + y + " ");  }  public static void main(String args[])  {  s run = new s();  Thread t1 = new  Thread(run); Thread t2 =  new Thread(run); t1.start();  t2.start();  }}</p>	Remembering	5,6
----	--	-------------	-----

**UNIT-IV**

**Part – A (Short Answer Questions)**

1	<b>Define</b> collections?	Remembering	7,8
2	<b>Define</b> Java collection Frame work.	Remembering	7,8
3	<b>Define</b> Array <b>List</b> with syntax	Remembering	7,8
4	<b>Define</b> Vector with syntax.	Remembering	7,8
5	<b>Define</b> hash table with syntax.	Remembering	7,8
6	<b>Define</b> stack with syntax.	Remembering	7,8
7	<b>Define</b> enumeration with syntax.	Remembering	7,8
8	<b>What</b> is Iterator?	Remembering	7,8
9	<b>Explain</b> the function of StringTokenizer.	Understanding	7,8
10	<b>Define</b> random class.	Remembering	7,8
11	<b>Define</b> Scanner class.	Remembering	7,8
12	<b>Define</b> Calendar class.	Remembering	7,8
13	<b>Define</b> Properties class	Remembering	7,8
14	<b>Define</b> Stream?	Remembering	7,8
15	<b>Define</b> byte stream?	Remembering	7,8
16	<b>Define</b> character stream?	Remembering	7,8

17	<b>Define</b> text input/output file?	Remembering	7,8
18	<b>Define</b> JDBC?	Remembering	7,8
<b>Part – B (Long Answer Questions)</b>			
1	<b>Explain</b> the Java Collection frame work with an example.	Understanding	7,8
2	<b>What is</b> ArrayList? Explain with an example.	Remembering	7,8
3	<b>Explain</b> briefly about Vector class with an example.	Understanding	7,8
4	<b>What is</b> hash table? Explain with an example?	Remembering	7,8
5	<b>Explain</b> about stack class with an example.	Understanding	7,8
6	<b>What is</b> enumeration? Explain about with an example.	Remembering	7,8
7	<b>Explain</b> briefly about iterator with an example.	Understanding	7,8
8	<b>What is</b> StringTokenizer? Explain with an example.	Remembering	7,8
9	<b>Explain</b> in detail about Random class with an example.	Understanding	7,8
10	<b>What is</b> Scanner class? Explain with an example.	Remembering	7,8
11	<b>Explain</b> in detail Calender class with an example.	Understanding	7,8
12	<b>What are</b> text input/output file operations? Explain	Remembering	7,8
13	<b>Explain</b> binary input/output file operations with examples.	Understanding	7,8
14	<b>What are</b> random access file operations?	Remembering	7,8
15	<b>Explain</b> briefly about File management using File class.	Understanding	7,8
16	<b>Distinguish</b> between a)InputStream and Reader classes b)OutputStream and Writer Classes	Understanding	7,8
17	<b>Explain</b> different types of JDBC drivers with diagrams.	Understanding	7,8
18	<b>Develop</b> a JDBC application for querying the database and processing the results.	Applying	7,8
19	<b>Develop</b> a JDBC application for updating data.	Applying	7,8
<b>Part – C (Problem Solving and Critical Thinking Questions)</b>			
1	<b>What</b> will be the output of the program? <pre> package foo; import java.util.Vector; private class MyVector extends Vector { int i = 1; public MyVector() { i = 2; } } public class MyNewVector extends MyVector { public MyNewVector () { i = 4; } public static void main (String args []) { MyVector v = new MyNewVector(); } } </pre>	Remembering	7,8
2	<b>Analyze</b> the following program and find the output of the program? <pre> public class Test { public static void main(String[] args) { System.out.println(Math.min(Double.MIN_VALUE, 0.0d));  } } </pre>	Analyzing	7,8
3	<b>Find</b> the output of the program. <pre> import java.util.*; </pre>	Remembering	7,8

	<pre> class I { public static void main (String[] args) { Object i = new ArrayList().iterator(); System.out.print((i instanceof List)+","); System.out.print((i instanceof Iterator)+","); System.out.print(i instanceof ListIterator); } } </pre>		
4	<p><b>Compare</b> the output before adding and after adding?</p> <pre> import java.util.*; public class ArrayListExample { public static void main(String args[]) { ArrayList&lt;String&gt; obj = new ArrayList&lt;String&gt;(); obj.add("Ajeet"); obj.add("Harry"); System.out.println("Currently the array list has following  elements:"+obj); obj.add(0, "Rahul"); obj.add(1, "Justin"); obj.remove("Chaitanya"); obj.remove("Harry"); System.out.println("Current array list is:"+obj); obj.remove(1); System.out.println("Current array list is:"+obj); } } </pre>	Remember	7,8
5	<p><b>What</b> is the output of the following program</p> <pre> import java.util.*; public class ArrayListExample { public static void main(String args[]) { ArrayList&lt;String&gt; a1 = new ArrayList&lt;String&gt;(); a1.add(1); a1.add(2); a1.add(3); a1.add(4); System.out.println("Current array list is:"+a1); Integer ia[]= new Integer [a1.size()]; ia= a1.toArray(ia); int sum=0; for (int i : ia) sum += i; System.out.println("Sum is:"+sum) } } </pre>	Remembering	7,8
6	<p><b>Analyze</b> the program and explain the importance of line 1 and give the output</p> <pre> import java.util.StringTokenizer; class Stdemo { Static String in= "title= java : the complete reference" +  "author= schildt" + "publisher= Osborne/mcgraw-hill " + "copyright= 2005"; </pre>	Analyzing	7,8



	<pre> public static void main(String args[]) { String key= st.nextToken(); ----- 1 String val= st.nextToken(); System.out.println( key + "\t" + val); } } </pre>		
7	<p><b>Analyze</b> the output of the below program?</p> <pre> import java.io.*; class BRRead { public static void main(String args[]) throws IOException { char c; { BufferedReader br= new BufferedReader( new InputStreamReader (System.in)); System.out.println("enter characters, „q“ to quit."); Do { c= (char) br.read(); System.out.println(c); While(c != „q“); } } } </pre>	Remembering	7,8
8	<p><b>Analyze</b> the program and give the output and also explain the importance of hasNext() and next() methods.</p> <pre> import java.util.*; class Findlinedemo { public static void main(String args[]) { String s="Name: Tom Age: 28 Id: 77"; Scanner sc= new Scanner(s); sc.findInLine ("Age"); if(sc.hasNext()) Syste.out.println(sc.next()); else Syste.out.println("error"); } } </pre>	Analyze	7,8
9	<p><b>What</b> is the output of the below program?</p> <pre> import java.io.*; class HTdemo { </pre>	Remembering	7,8

	<pre> public static void main(String args[]) {     Hashtable&lt; String, Double&gt; balance= Hashtable&lt; String, Double&gt;();      Enumeration&lt;String&gt; names;     String str;     double bal;     balance.put("abc" 3435.35);     balance.put("abc" 1254.35);     names= balance.keys();     while(names.hasMoreElements())     {         Str= names.nextElement();         System.out.println(str + " : " + balance.get(str);     }     System.out.println(); } </pre>		
10	<p><b>Analyze</b> the below program and give the output of the following program?</p> <pre> import java.util.Scanner; class Division { public static void main(String[] args) { int a, b, result; Scanner input = new Scanner(System.in); System.out.println("Input two integers"); a = input.nextInt(); b = input.nextInt(); result = a / b; System.out.println("Result = " + result); } </pre>	Analyze	7,8
<b>UNIT-V</b>			
<b>Part - A (Short Answer Questions)</b>			
1	<b>Define</b> AWT class hierarchy?	Remembering	9,10
2	<b>Distinguish</b> between swings Vs AWT?	Analyzing	9,10
3	<b>Explain</b> the hierarchy for swing?	Understanding	9,10
4	<b>Define</b> components?	Remembering	9,10
5	<b>Define</b> containers?	Remembering	9,10
6	<b>Define</b> JFrame, JApplet, JDialog and Jpanel?	Remembering	9,10
7	<b>Define</b> some of swing components?	Remembering	9,10
8	<b>Define</b> Jbutton, JLabel, JTextField and JtextArea?	Remembering	9,10
9	<b>Define</b> Layout management?	Remembering	9,10
10	<b>List</b> Layout manager types – border and grid flow?	Remembering	9,10
11	<b>Explain</b> about Events, Event sources, Event classes	Understanding	9,10
12	<b>Explain</b> about Event Listeners	Understanding	9,10
13	<b>Compare</b> and contrast the relationship between Event sources and Listeners?	Understanding	9,10
14	<b>Define</b> Delegation event model?	Remembering	9,10
15	<b>Explain</b> various events for handling a button click?	Understanding	9,10
16	<b>Explain</b> various events for handling mouse events?	Understanding	9,10
17	<b>Define</b> adapter class?	Remembering	9,10

18	<b>Distinguish</b> between applet and application?	Analyzing	9,10
19	Explain the life cycle of an Applet.	Understanding	9,10
20	<b>Describe</b> applet security issues?	Understand	9,10
<b>Part - B (Long Answer Questions)</b>			
1	<b>Explain</b> in detail about hierarchy for swing?	Understanding	9,10
2	<b>Explain</b> in detail about hierarchy for awt?	Understanding	9,10
3	<b>Explain</b> in detail about Layout management?	Understanding	9,10
4	<b>Develop</b> a java program for handling a button clicks?	Applying	9,10
5	<b>Develop</b> a java program for handling mouse Related events?	Applying	9,10
6	<b>Explain</b> in detail about Events, Event sources and Event classes	Understanding	9,10
7	<b>Explain</b> in detail about Event sources and Listeners	Understanding	9,10
8	<b>Develop</b> a java program for simple applet?	Applying	9,10
9	<b>Develop</b> a java program for passing parameters to applet?	Applying	9,10
10	<b>Develop</b> an applet that receives an integer in one text field and computes its factorial value and returns it in another text field, when the button named "compute" is clicked	Applying	9,10
11	<b>Explain</b> briefly about Adapter classes.	Understanding	9,10
12	<b>What</b> is the importance of Delegation Event Model on Event Handling	Remembering	9,10
13	<b>Distinguish</b> various differences between Swing and AWT	Analyzing	9,10
14	<b>Explain</b> the differences between applets and applications	Understanding	9,10
15	<b>Explain</b> various swing components in detail	Understanding	9,10
<b>Part – C (Problem Solving and Critical Thinking Questions)</b>			
1	<b>Predict</b> the output using below code and what happens if below code is not used in GUI programs? <pre>public void windowClosing(WindowEvent e) { dispose(); System.exit(0);</pre>	Understand	9,10
2	<b>Explain</b> the below code and mention its important <pre>Public void actionPerformed(ActionEvent e) { it(e.getSource()== b1)---Assume b1 is variable of button  { int x= Integer.parseInt(t1.getText()) ---- Assume t1 is variable of textfield int y= Integer.parseInt(t2.getText()) ---- Assume t2 is variable of textfield int sum= X+Y; t3.setText(""+sum); ---- Assume t3 is variable of textfield } }</pre>	Understanding	9,10
3	<b>What</b> is the output the program <pre>import java.awt.*; import java.swing.*; /* &lt; applet code= "JLabeldemo" width= 200 height =200&gt;  &lt;/applet&gt; */ public class JLabeldemo extends JApplet { ImageIcon i= new ImageIcon(" india.gif"); JLabel ji=new JLabel(" INDIA", I, JLabel.CENTER);</pre>	Remembering	9,10

	<pre> add(ji); } } </pre>		
4	<p><b>Explain</b> the output of the following program?</p> <pre> import java.applet.*; import java.awt.*; public class Main extends Applet{  public void paint(Graphics g){     g.drawString("Welcome in Java Applet.",40,20);  } } &lt;HTML&gt; &lt;HEAD&gt; &lt;/HEAD&gt; &lt;BODY&gt; &lt;div &gt; &lt;APPLET CODE="Main.class" WIDTH="800" HEIGHT="500"&gt;  &lt;/APPLET&gt; &lt;/div&gt; &lt;/BODY&gt; &lt;/HTML&gt; </pre>	Evaluated	9,10
5	<p>Explain the usage of the following code?</p> <pre> public void actionPerformed(ActionEvent ae){     try{         num = Integer.parseInt(input.getText());         sum = sum+num;         input.setText("");         output.setText(Integer.toString(sum));         lbl.setForeground(Color.blue);         lbl.setText("Output of the second Text Box : "  + output.getText());     }     catch(NumberFormatException e){         lbl.setForeground(Color.red);         lbl.setText("Invalid Entry!");     } } </pre>	Understanding	9,10
6	<p><b>Analyze</b> the program and explain the importance 1 to 4 line and give the output.</p> <pre> import java.awt.*; class Frame1 extends Frame { Frame1() { setTitle("demo");----- 1 setSize(200,200); setVisible(true); ----- 2 setLayout(new FlowLayout()); Label l1= new Label("java"); Label l2= new Label("j2ee"); add(l1); ----- 3 add(l2); ----- 4 } } </pre>	Analyze	9,10

	<pre> } } Class Labeldemo { Public static void main(String args()); { Frame1 f= new Frame(); } } </pre>		
7	<p><b>What</b> is the output the program</p> <pre> import java.awt.*; import java.applet.*; /* &lt; applet code= "statusdemo.class" width= 200 height =200&gt;  &lt;/applet&gt; */ public class satusdemo extends Applet { Public void init() { setBackground(Color.red); } Public void paint(Graphics g) { g.drawString("this is in the applet window" 10,20) showStatus("this is the status window message"); } } </pre>	Remembering	9,10
8	<p><b>Explain</b> the usage of following line of code line by line.</p> <pre> Public void mouseClicked(MouseEvent me) { Mousex-=0; Mousey=10; Msg= "mouse clicked" Repaint(); } Public void mouseEntered(MouseEvent me) { Mousex-=0; Mousey=10; Msg= "mouse entered" Repaint(); } </pre>	Understanding	9,10
9	<p><b>What</b> is the output of the program</p> <pre> import java.applet.Applet; import java.awt.*; public class Sms extends Applet { public void init() { } public void paint(Graphics g) { g.setColor(Color.blue); Font font = new Font("verdana", Font.BOLD, 15); g.setFont(font); g.drawString("Welcome To Aeronautical Eng College", 50, 50); } } </pre>	Remembering	9,10

	}		
10	<p><b>What</b> is the output the program and explain line 1 and 2.</p> <pre> import java.awt.*; import java.applet.*; /* &lt; applet code= "GridLayoutDemo" width= 300 height =300&gt;  &lt;/applet&gt; */ public class GridLayoutDemo extends Applet { static final int n=5; public void init() { setLayout(new GridLayout(      n , n)); -----1 setFont (new Font ("SamsSerof", Font.BOLD, 24)); --- 2  for (int j=0l j&lt;n; j++) { int k= I * n + j; if(k&gt;00) Add( new button (" " + k0; }}}} </pre>	Remembering	9,10

Prepared by : Ms. K. Radhika Assistant Professor  
Mr N V. Krishna Rao, Assistant Professor, CSE  
Mr. P .Ravinder , Assistant Professor, CSE

Date : 05 December, 2016

**HOD, CSE**