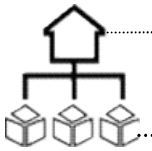


OBJECT ORIENTED PROGRAMMING

1. Which one is pure object oriented language? *[BREB(AJE)-2019]*

a) C++ b) C+ c) JAVA d) None **Ans.:** d

Explanation:

- **Pure Object Oriented Language**
  A language is called PURE object oriented if It contains only objects and classes. In it we do access everything through 'Message Passing'.
  Example*:*SmallTalk
- **JAVA** is not a 'Pure Object Oriented Language' as in contains Primitive Data Types (int, long, double) and wrapper class.
- **Fully Object Oriented Language**
  A language is called FULLY object oriented if it contains all the fundamental features of object oriented programming and it can have primitive data types or not.
  Some of fully object oriented languages are as follows:
  - ✓ JAVA
  - ✓ C#
  - ✓ Visual Basic
  - ✓ C++

2. Which was the first purely object oriented programming language developed?

a) Java b) C++ c) SmallTalk d) Kotlin **Ans.:** c

*Explanation:*SmallTalk was the first programming languagedeveloped which was purely object oriented. It wasdeveloped by Alan Kay. OOP concept came into picture in**1970's.**

3. C++ was originally developed by

a) Clocksin and Mellish b) Donald E. Knuth
c) Sir Richard Hadlee d) BjameStroustrup **Ans.:** d

Explanation:**C++ was developed** by Danish computer scientist **Bjarne Stroustrup** at Bell Labs since 1979 as an extension of the C language;

4. Which of the followinig are procedural languages?

a) Pascal b) Smalltalk c) C d) Both(a) and (c) **Ans.:** d

*Explanation:*Procedural languages sequentially execute a set of imperative statements to achieve the desired
effect. Most of the traditional languages fall in this category.

5. Which one is a feature of Object oriented programming? *[ANE-BPSC-2019]*

a) Polymorphism b) Friend function c) Structure d) loop **Ans.:** a

Explanation:

- **What is OOP?**
  OOP stands for Object-Oriented Programming. Procedural programming is about writing procedures or methods that perform operations on the data, while object-oriented programming is about creating objects that contain both data and methods.
- **Object-oriented programming has several advantages over procedural programming:**
  - ✓ OOP is faster and easier to execute
  - ✓ OOP provides a clear structure for the programs
  - ✓ OOP helps to keep the Java code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug
  - ✓ OOP makes it possible to create full reusable applications with less code and shorter development time
- **What is a dry principle?**
  The "Don't Repeat Yourself" (DRY) principle is about reducing the repetition of code. You should extract out the codes that are common for the application, and place them at a single place and reuse them instead of repeating it.

6. Which one is not the OOP language?

a) Java b) C c) PhP d) C++ **Ans.:** b

7. In Java, which operator is used to create an object ---*[ICB(AP)-2017]*

a) class b) new
c) print d) None of the above **Ans.:** b

Explanation: java creating a object like: Class_Name object_Name=new Class_Name()

class	object
Fruit	apple banana orange

So, a class is a template for objects, and an object is an instance of a class.

- **A class in Java can contain:**
  - ✓ Fields
  - ✓ Methods
  - ✓ Constructors
  - ✓ Blocks
  - ✓ Nested class and interface

8. Which of the following is a valid declaration to create an object of class Box?

[Com(AP)- 2019]

a) Box obj = new Box(); b) Box obj =new Box;
c) obj = new Box(); d) new Box obi; **Ans.:** a

9. How many objects can be declared of a specific class in a single program?

a) 32768 b) 1

c) as many as you want d) None **Ans.:** c

Explanation: A simple java program for creating an object and calling a variable.

```
public class ClassObject {
int value=5;

public static void main(String[] args) {
ClassObject obj=new ClassObject();
System.out.println(obj.value);
}
}
```

10. Does constructor overloading include different return types for constructors to be overloaded? *[Combined(SO-IT/ICT)-2019]*

a) Yes, if return types are different, signature becomes different
b) Yes, because return types can differentiate two functions
c) No, return type can't differentiate two functions
d) No, constructors doesn't have any return type **Ans.:** d

Explanation:

- **Constructors in Java**

In Java, a constructor is a block of codes similar to the method. It is called when an instance of the class is created. At the time of calling constructor, memory for the object is allocated in the memory.

It is a special type of method which is used to initialize the object.Every time an object is created using the new () keyword, at least one constructor is called.

- **Rules for creating Java constructor**

There are two rules defined for the constructor.

1. Constructor name must be the same as its class name
2. A Constructor must have no explicit return type

- **Types of Java constructors**

There are two types of constructors in Java:

1. Default constructor (no-arg constructor)
2. Parameterized constructor

11. Public class Test{} what is the prototype of the default constructor? *[BREB-2019]*

a) Test() b) Test(Void)
c) public Test() d) Public Test(Void) **Ans.:** c

12. When the compiler cannot differentiate between two overloaded constructors, they are called? *[BREB-2019]*

a) Overloaded b) destructed c) ambiguous d) dubious **Ans.:** c

13. Which of the following is the destructor of class Vehicle? *[Combined(O-IT/ICT)-2019]*

a) *Vehicle0 b) ~Vehicle()

c) ~Vehicle(int value) d) *Vehicle(int value) **Ans.:** b

Explanation: **C++ destructor** is a special member function that is executed automatically when an object is destroyed that has been created by the constructor. C++ destructors are used to de-allocate the memory that has been allocated for the object by the constructor.

Can there be more than one destructor in a class?

No, there can only one destructor in a class with classname preceded by ~, no parameters and no return type.

14. Which operator is used to declare the destructor in C++? *[Combined(SO-IT/ICT)-2018]*

a) # b) ~ c) @ d) – **Ans.:** b

The Objects Oriented programming language supports all the features of normal programming languages. In addition it supports some important concepts and terminology which has made it popular among programming methodology.

15. The important features of Object oriented programming are:

- ✓ Inheritance
- ✓ Polymorphism
- ✓ Data Hiding
- ✓ Encapsulation
- ✓ Overloading
- ✓ Reusability

☞ **Encapsulation** is a feature of oop languages that provides the concept of wrapping of data and member function in single unit.

☞ **Inheritance** is a concept that provide reusability of codes and recourse in object oriented programming languages

☞ **Polymorphism** this is existing features of oop. It provides that concept of same name or different behavior.

☞ **Data abstraction** This features of oop languages focus on hiding of complexity of program.

16. Which is not feature of OOP in general definitions?

a) Code reusability b) Modularity

c) Duplicate/Redundant data d) Efficient Code **Ans.:** c

17. Which Feature of OOP illustrated the code reusability?

a) Polymorphism b) Abstraction c) Encapsulation d) Inheritance **Ans.:** d

18. Which of the two features match each other?

a) inheritance and Encapsulation b) Encapsulation and Polymorphism

c) Encapsulation and Abstraction d) Abstraction and Polymorphism **Ans.:** c

19. If same message is passed to objects of several different classes and all of those can respond in a different way, what is this feature called? *[Combined(SO-IT/ICT)-2018]*

a) Inheritance b) Overloading c) Polymorphism d) Overriding **Ans.:** c

20. Which of the statements are true?

I. Function overloading is done at compile time.

II. Protected members are accessible to the member of derived class.
III. A derived class inherits constructors and destructors.
IV. A friend function can be called like a normal function.
V. Nested class is a derived class.

a) I, II, III b) II,III,V c) III,IV,V d) I,II,IV **Ans.:** d

21. When the compiler cannot differentiate between two overloaded constructors, they are called

a) Overloaded b) Destructed c) Ambiguous d) Dubious **Ans.:** c

BASIC OF JAVA

- **Inheritance in Java**

  The process by which one class acquires the properties (data members) and functionalities (methods) of another class is called **inheritance**. The aim of inheritance is to provide the reusability of code so that a class has to write only the unique features and rest of the common properties and functionalities can be extended from the another class**. Inheritance represents the** IS-A relationship **which is also known as a *parent-child* relationship.**

- **Child Class:**

  The class that extends the features of another class is known as child class, sub class or derived class.

- **ParentClass:**

  The class whose properties and functionalities are used **(inherited) by** another class is known as parent class, super class or Base class.

Syntex of Inheritance

```
class XYZ extends ABC
{
}
```

- **Full Example of Inheritance:**

```
class Employee{// parent class
  float salary=2000;
}
class Teacher extends Employee{// Child class
  int bonus=1000;
 public static void main(String args[]){
  Teacher t=new Teacher();
 System.out.println("Teachersalary is:"+t.salary);
```

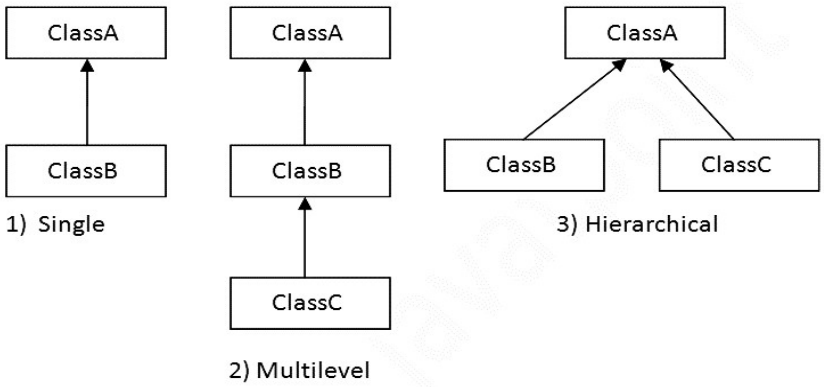
```
System.out.println("Bonus of Teacheris:"+t.bonus);
  }
}
```

- **Types of inheritance**

Single Inheritance: Refers to a child and parent class relationship where a class extends the class.

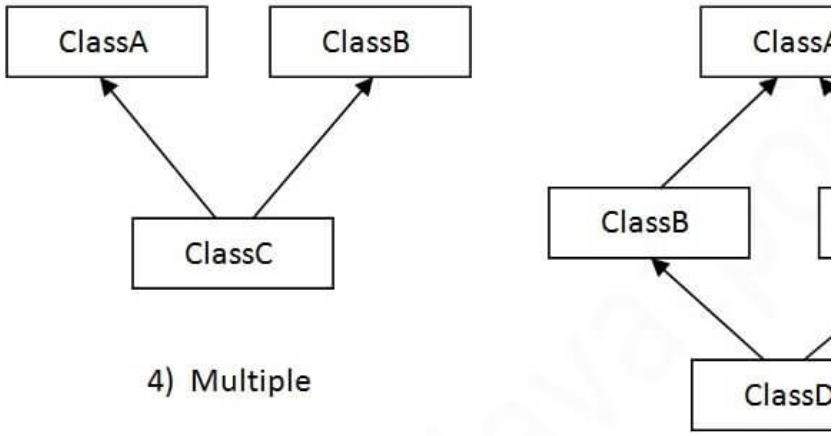
Multilevel inheritance: Refers to a child and parent class relationship where a class extends the child class. For example class C extends class B and class B extends class A.

Hierarchical inheritance: Refers to a child and parent class relationship where more than one classes extends the same class. For example, classes B, C extends the same class A.



Multiple Inheritances: refers to the concept of one class extending more than one classes, which means a child class has two parent classes. For example class C extends both classes A and B.

Hybrid inheritance: Combination of more than one types of inheritance in a single program. For example class A & B extends class C and another class D extends class A then this is a hybrid inheritance example because it is a combination of single and hierarchical inheritance.



Note: **Multiple inheritances are not supported in Java through class.**

22. Which among the following best describes the Inheritance?

a) Copying the code already written
b) Using the code already written once
c) Using already defined functions in programming language
d) Using the data and functions into derived segment **Ans.:** d

23. How many basic types of inheritance are provided as OOP feature?

a) 4 b) 3 c) 2 d) 1 **Ans.:** a

Explanation: There are basically 4 types of inheritance provided in OOP, namely, single level, multilevel, multiple and hierarchical inheritance. We can add one more type as Hybrid inheritance but that is actually the combination any types of inheritance from the 4 basic ones.

24. Which among the following best defines single level inheritance?

a) A class inheriting a derived class
b) A class inheriting a base class
c) A class inheriting a nested class
d) A class which gets inherited by 2 classes **Ans.:** b

25. Which of the keywords can be used in a subclass to call the constructor of super class?

[Combined(AP)-2018]

a) Extent b) Extends c) Super d) This **Ans.:** c

26. Which programming language doesn't support multiple inheritance?

a) C++ and Java b) C and C++
c) Java and SmallTalk d) Java **Ans.:** d

27. Members which are not intended to be inherited are declared as _____

a) Public members b) Protected members
c) Private members d) Private or Protected members **Ans.:** c

28. If a derived class object is created, which constructor is called first?

a) Base class constructor b) Derived class constructor

c) Depends on how we call the object d) not possible **Ans.:** a

Explanation: Base class constructors are always called in the derived class constructors. Whenever you create derived class object, first the base class default constructor is executed and then the derived class's constructor finishes execution.

Points to remember that, the constructor of base class is:

- ✓ default or non-parameterized constructor or
- ✓ parameterized constructor

If it is default or non - parameterized constructor then there is nothing to do because compiler itself generates a call to base class constructor.But if it is parameterized constructor then it is programmer's responsibility to call base class constructor. It can be done from derive class constructor using **super** keyword (in Java) in the **first line** of the derive class constructor.

29. Which type of inheritance cannot involve private inheritance?

a) Single level b) Multiple

c) Hybrid d) All types can have private inheritance **Ans.:** d

➢ **Java Polymorphism**

Method Overloading in Java

If a class has multiple methods having same name but different in parameters, it is known as **Method Overloading**. If we have to perform only one operation, having same name of the methods increases the readability of the program.

Different ways to overload the method

There are two ways to overload the method in java

1. By changing number of arguments
   - ✓ Like: add(int,int)
   - ✓ add(int,int,int)
2. By changing the data type
   - ✓ add(int,int)
   - ✓ add(int,float)
3. Sequence of Data type of parameters.
   - ✓ add(int,float)
   - ✓ add(float,int)

In Java, Method Overloading is not possible by changing the return type of the method only.

✍ **Points to Note:**
1. Static Polymorphism is also known as compile time binding or early binding.
2. Static binding happens at compile time. Method overloading is an example of static binding where binding of method call to its definition happens at Compile time.

30. What is the process of defining two or more methods within the same class that have same name but different parameters declaration? *[Combined(AP)-2018]*

a) Method overriding b) Method overloading
c) Method hiding d) Method duplicating **Ans.:** b

Method overriding in java

Declaring a method in **sub class** which is already present in **parent class** is known as method overriding. Overriding is done so that a child class can give its own implementation to a method which is already provided by the parent class. In this case the method in parent class is called overridden method and the method in child class is called overriding method.

Rules for Java Method Overriding

- ✓ The method must have the same name as in the parent class
- ✓ The method must have the same parameter as in the parent class.
- ✓ There must be an IS-A relationship (inheritance).

Method overriding is used for runtime polymorphism

Example:

```
class Vehicle{
void run(){System.out.println("Vehicle is running");}
     }
 class Bike2 extends Vehicle{
 void run(){System.out.println("Bike is running safely");}
 public static void main(String args[]){
 Bike2 obj = new Bike2();
     obj.run();
}}
```

Ans.: Bike is running safely

➢ **Method Overriding and Dynamic Method Dispatch**

Method Overriding is an example of **runtime polymorphism.** When a parent class reference points to the child class object then the call to the overridden method is determined at runtime, because during method call which method(parent class or child class) is to be executed is determined by the type of object. This process in which call to the overridden method is resolved at runtime is known as dynamic method dispatch.

```
Class Test{
public void exam()
{
    System.out.println("exam() method of parent class");
}
}
Class Demo extends Test {
public void exam(){
    System.out.println("exam() method of Child class");
}
public void newMethod(){
    System.out.println("new method of child class");
}
public static void main(String args[]){
    Test obj =new Test ();
    obj. exam (); // Called parent class object.
    Test obj2=new Demo();
    obj2.disp();
}
}
```

Output: exam() method of parent class
exam()method of Child class

*Explanation:*Test obj2=new Demo(). When parent class reference refers to the child class object then the overriding method (method of child class) is called. This is called dynamic method dispatch and runtime polymorphism

Points to Note:

Can we override static method?

No, a static method cannot be overridden. It can be proved by runtime polymorphism

Why can we not override static method?

It is because the static method is bound with class whereas instance method is bound with an object. Static belongs to the class area, and an instance belongs to the heap area.

Can we override java main method?

No, because the main is a static method.

31. Which of these keywords can be used to prevent Method overriding?

a) static b) constant c) protected d) final **Ans.:** d

- ✓ The **final** keyword can be used with class method and variable. A final class cannot be instantiated, a final method cannot be overridden and a final variable cannot be reassigned.
- ✓ The **finally** keyword is used to create a block of code that follows a try block. A finally block of code always executes, whether or not an exception has occurred. Using a finally block allows you to run any cleanup-type statements that you just wish to execute, despite what happens within the protected code.
- ✓ The **finalize ()** method is used just before object is destroyed and can be called just prior to object creation.

32. Which of these can be overloaded?

a) Methods b) Constructors

c) All of the mentioned d) None of the mentioned **Ans.:** c

33. How to access the overridden method of base class from the derived class? *[Combined(SO-IT/ICT)-2018]*

a) Using arrow operator b) Using dot operator

c) Using scope resolution operator d) Can't be accessed once overridden **Ans.:** b

34. Which of these is correct way of calling a constructor having no parameters, of superclass A by subclass B?

a) super(void); b) superclass.() c) super.A() d) super() **Ans.:** d

35. Which of these is supported by method overriding in Java?

a) Abstraction b) Encapsulation

c) Polymorphism d) None of the mentioned **Ans.:** c

- ➢ **Java Package**: A **java package** is a group of similar types of classes, interfaces and sub-packages. Package in java can be categorized in two form, built-in package and user-defined package. There are many built-in packages such as java, lang, awt, javax, swing, net, io, util, sql etc.
- ➢ **Advantage of Java Package**
  - ✓ Java package is used to categorize the classes and interfaces so that they can be easily maintained.
  - ✓ Java package provides access protection.
  - ✓ Java package removes naming collision.

36. Which of the following is an incorrect statement about packages? *[Combined(SO-IT/ICT)-2018]*

a) Package defines a namespace in which classes are stored

b) A package can contain other packages within

c) Java uses file system directories to store packages

d) A package can be renamed without renaming the directory, in which the classes are stored

Ans.: d

- **Exception Handling in Java**

The **Exception Handling in Java** is one of the powerfulmechanisms to handle the **runtime errors**so that normal flow of the application can be maintained.Exception is an abnormal condition. In Java, an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.

try	The "try" keyword is used to specify a block where we should place exception code. The try block must be followed by either catch or finally. It means, we can't use try block alone.
catch	The "catch" block is used to handle the exception. It must be preceded by try block which means we can't use catch block alone. It can be followed by finally block later.
finally	The "finally" block is used to execute the important code of the program. It is executed whether an exception is handled or not.
throw	The "throw" keyword is used to throw an exception.
throws	The "throws" keyword is used to declare exceptions. It doesn't throw an exception. It specifies that there may occur an exception in the method. It is always used with method signature.

Note: The **finally** block is used to execute a given set of the statement whether the exception is thrown or not. *[SO(IT/ICT)-2020]*

37. When Exceptions in Java doesarise in code sequence?

a) Run Time b) Compilation Time
c) Can Occur Any Time d) None of the mentioned **Ans.:** a

38. Which of these keywords is not a part of exception handling?

a) try b) finally c) thrown d) catch **Ans.:** c

39. Which of these keywords must be used to monitor for exceptions?

a) try b) finally c) throw d) catch **Ans.:** a

40. Which of the following is a super class of all exception type classes?

a) Catchable b) Runtime Exceptions
c) String d) Throwable **Ans.:** d

OOP PREVIOUS YEAR QUESTIONS

41. The operator that cannot be overloaded is --- *[Combined(O-IT/ICT)-2019]*

a) ++ b) () (c) ~ d) :: **Ans.:** d

42. Which of the following is the destructor of class Vehicle? *[Combined(O-IT/ICT)-2019]*

a) *Vehicle0 b) ~Vehicle()

c) ~Vehicle(int value) d) *Vehicle(int value) **Ans.:** b

43. Which of the following operator functions cannot be global i.e must be a member function? *[Combined(O-IT/ICT)-2019]*

a) Conversion operator b) new

c) delete d) all of these **Ans.:** a

44. Which of the following is not an operator in Java? *[Combined(O-IT/ICT)-2019]*

a) instanceof b) sizeof c) new d) All of this **Ans.:** b

45. Find the output of following Java code line: *[Combined(O-IT/ICT)-2019]*

System.out.println(math.floor(-7.4)

a) -7 b) -7.4 c) -8 d) -7.2 **Ans.:** c

46. Which functions overload the >> operator?*[Combined(O-IT/ICT)-2019]*

a) gt() b) more() c) ge() d) None of this **Ans.:** d

*Explanation:*rshift() overloads the >> operator

47. Which of these is not a core data type? *[Combined(O-IT/ICT)-2019]*

a) Lists b) Dictionary c) Class d) Tuples+95 **Ans.:** c

48. Which of the following provides a programmer with the facility of using object of a class inside other classes? *[PKB(programmer)-2019]*

a) Inheritance b) Abstraction c) Encapsulation d) Composition **Ans.:** d

49. In C++,the idea to hiding the details of how something is implemented is known as *[PKB(AP)-2019]*

a) inheritance b) encapsulation c) recursion d) polymorphism **Ans.:** b

50. In Java, which operator is used to create an object *[PKB(AP)-2019]*

a) class b) scanf c) print d) None of these **Ans.:** d

51. Which one is pure object oriented language?*[BREB(AJE)-2019]*

a) C++ b) C+ c) JAVA d) None **Ans.:** d

52. Which operator is used to declare the destructor in C++? *[Combined(SO-IT/ICT)-2018]*

a) # b) ~ c) @ d) $ **Ans.:** b

53. Which alternative can replace the throw statement in c++?*[Combined(SO-IT/ICT) 2018]*

a)for b) break c) return d) exit **Ans.: c**

54. What type of reference should be used in vector arithmetic in C++? *[Combined(SO-IT/ICT)-2018]*

a) Dynamic b) const

c) a and b d) none of the mentioned **Ans.:** b

55. Which of these data types is used by operating system to manage the Recursion in Java? *[Combined(SO-IT/ICT)-2018]*

a) Array b)Stack c) Queue d) Tree

56. Which of the following is an incorrect statement about packages? *[Combined(SO-IT/ICT)-2018]*

a)Package defines a namespace in which classes are stored

b)A package can contain other packages within

c) Java uses file system directories to store packages

d)A package can be renamed without renaming the directory, in which the classes are stored **Ans.:** d

57. A derived class inherits attributes from a _ _ *[Combined(SO-IT/ICT)-2018]*

a) Super Class b) Sub Class c) Inner Class d) Upper Class

58. Multiple inheritances in Java can be implemented using which of the following? *[Combined(SO-IT/ICT)-2018]*

a) Interfaces b) Multithreading c) Protected methods d) Private methods **Ans.:** a

59. Which component is used to compile, Debug and execute java program? *[Combined(SO-IT/ICT)-2018]*

a) JVM b) JDK c) JIT d) JRE**Ans.: b**

60. Object being passed to a copy constructor------*[Combined(SO-IT/ICT)-2018]*

a) Must be passed by reference b) Must be passed by value

c) Must be passed with integer type d) Must not be mentioned in parameter list **Ans.:** a

The mandatory to pass the object by reference. Otherwise the object will try to create another object to copy its value, its turn a constructor will be called and this will keep on calling itself. This will cause the compiler to give out of memory error.

61. Why do we need to handle exceptions? *[Combined(SO-IT/ICT)-2018]*

a) To prevent abnormal termination of program

b) To encourage exception prone program

c) To avoid syntax errors

d) To save memory **Ans.:** a

62. If same message is passed to objects of several different classes and all of those can respond in a different way, what is this feature called? *[Combined(SO-IT/ICT)-2018]*

a) Inheritance b) Overloading c) Polymorphism d) Overriding **Ans.:** c

63. Does constructor overloading include different return types for constructors to be overloaded? *[Combined(SO-IT/ICT)-2018]*

a) Yes, if return types are different, signature becomes different

b) Yes, because return types can differentiate two functions

c) No, return type can't differentiate two functions

d) No, constructors doesn't have any return type**Ans.: d**

64. How to access the overridden method of base class from the derived class? *[Combined(SO-IT/ICT)-2018]*

a) Using arrow operator b) Using dot operator

c) Using scope resolution operator d) Can't be accessed once overridden **Ans.:** b

65. Source code is a(an) ---------- *[Combined(SO-IT/ICT)-2018]*

a) Program b) List of commands

c) Source of variables d) Explanation of the algorithm **Ans.:** a

66. Which of the keywords can be used in a subclass to call the constructor of superclass? *[Combined(AP)-2018]*

a)Extent b) Extends c) Super d) This **Ans.:** c

67. Multi Threaded programs are---------- *[Combined(AP)-2018]*

a) Lesser prone to deadlocks b) more prone to deadlocks
c) not at all prone to deadlock d) always results in deadlocks **Ans.:** b

68. Which of the following is a valid declaration titan object of class Box? *[Combined(AP)-2018]*

a) Box obj = new Box(); b) Box obj =new Box;
c) obj = new Box(); d) new Box obi; **Ans.:** a

69. What is the process of defining two or more methods within the same class that have same name but different parameters declaration? *[Combined(AP)-2018]*

a) Method overriding b) Method overloading
c) Method hiding d) Method duplicating **Ans.:** b

70. Which of the following format is a correct format for declaration of function? *[Combined(AP)-2018]*

a) return-type function-name(argument type);
b) return-type function-name(argument type){}
c) return-type (argument type) function-name;
d) return-type {} function-name **Ans.:** a

71. Which part of a class is invoked when an object is initialized in java? *[SBL,JBL (SO-IT/ICT)-2018]*

a) constructor b) fields c) methods d) class **Ans.:** a

72. Which is the keywords of structured programming? *[BB(AP)2016]*

a) Keywords b) const c) volatile d) Above all **Ans.:** a

73. The value 9.87 to 10 when use? *[BB(AP)2016]*

a) floor () b) ceil() c) both d) None **Ans.:** b

74. A class that is inherited in java is called a *[SBL(AP)2016]*

a) Subclass b) Superclass c) Static class d) Implement class **Ans.:** b