As per New CBCS Syllabus for Second Year, Third Semester,
B.Sc. Computers of All Universities in Andhra Pradesh w.e.f. 2015-2016

Object Oriented Programming in Java

- Sravya Lakshmi Malyala
- R. Kiran Kumar
- K. Anitha

Himalaya Publishing House
ISO 9001:2008 CERTIFIED

OBJECT ORIENTED PROGRAMMING IN JAVA

(As Per New CBCS Syllabus for Second Year Third Semester, B.Sc. Computers of All Universities in Andhra Pradesh w.e.f. 2015-2016)

SRAVYA LAKSHMI MALYALA
M.Sc. (Computer Science), M.Sc. (Mathematics),
M.Tech. (CSE), Canada.
Former Assistant Professor, Department of Computer Science,
Aditya Educational Institutions, Kakinada, India.

DR. R. KIRAN KUMAR
M.Tech., Ph.D.
HOD, Department of Computer Science,
Coordinator, Centre for Research Studies,
Krishna University, Machilipatnam - 521 001.

K. ANITHA
MCA, M.Tech. (CSE),
HOD, Department of Computer Science,
Sri Durga Malleswara Siddhartha Mahila Kalasala,
Bundur Road, Vijayawada.

Himalaya Publishing House
ISO 9001:2008 CERTIFIED

© **Authors**

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording and/or otherwise without the prior written permission of the publisher.

First Edition : 2017

Published by : Mrs. Meena Pandey for **Himalaya Publishing House Pvt. Ltd.,**
"Ramdoot", Dr. Bhalerao Marg, Girgaon, Mumbai - 400 004.
Phone: 022-23860170/23863863; **Fax:** 022-23877178
E-mail: himpub@vsnl.com; **Website:** www.himpub.com

Branch Offices :

New Delhi : "Pooja Apartments", 4-B, Murari Lal Street, Ansari Road, Darya Ganj,
New Delhi - 110 002. Phone: 011-23270392, 23278631; Fax: 011-23256286

Nagpur : Kundanlal Chandak Industrial Estate, Ghat Road, Nagpur - 440 018.
Phone: 0712-2738731, 3296733; Telefax: 0712-2721216

Bengaluru : Plot No. 91-33, 2nd Main Road Seshadripuram, Behind Nataraja Theatre,
Bengaluru - 560020. Phone: 08041138821; Mobile: 9379847017, 9379847005

Hyderabad : No. 3-4-184, Lingampally, Besides Raghavendra Swamy Matham, Kachiguda,
Hyderabad - 500 027. Phone: 040-27560041, 27550139

Chennai : New No. 48/2, Old No. 28/2, Ground Floor, Sarangapani Street, T. Nagar,
Chennai - 600 012. Mobile: 9380460419

Pune : First Floor, "Laksha" Apartment, No. 527, Mehunpura, Shaniwarpeth
(Near Prabhat Theatre), Pune - 411 030. Phone: 020-24496323/24496333;
Mobile: 09370579333

Lucknow : House No. 731, Shekhupura Colony, Near B.D. Convent School, Aliganj,
Lucknow - 226 022. Phone: 0522-4012353; Mobile: 09307501549

Ahmedabad : 114, "SHAIL", 1st Floor, Opp. Madhu Sudan House, C.G. Road, Navrang Pura,
Ahmedabad - 380 009. Phone: 079-26560126; Mobile: 09377088847

Ernakulam : 39/176 (New No. 60/251), 1st Floor, Karikkamuri Road, Ernakulam,
Kochi - 682011. Phone: 0484-2378012, 2378016; Mobile: 09387122121

Bhubaneswar : 5 Station Square, Bhubaneswar - 751 001 (Odisha).
Phone: 0674-2532129; Mobile: 09338746007

Kolkata : 108/4, Beliaghata Main Road, Near ID Hospital, Opp. SBI Bank,
Kolkata - 700 010. Phone: 033-32449649; Mobile: 7439040301

DTP by : Sneha Gaonkar

Printed at : M/s. Seven Hills Printers, Hyderabad. On behalf of HPH.

PREFACE

In today's competitive business world, organizations are struggling to stand first and get more profits. Information technology is at the first position for long years with many innovative trends. It has multiple stream and platforms to develop an application or product. When we talk of programming languages and technologies, Java is flexible and reliable to develop applications. The usage of Java programming language has increased tremendously.

Java provides a great flexibility while choosing a platform. By using Java, we can develop a code for web and mobile applications. It is one of the best programming languages for networking of computers. It is very efficient in making applications in a short period of time. Java is the most flexible platform, to develop several applications for the system, embedded devices like mobile, laptops, tablets and many more. There are so many popular applications for mobile phones which are developed using Java programming language.

It is important for information technology industry to develop and create multiple web-based or server-based applications to enhance the industrial competency. There is huge scope for this programming language as Java is a robust, multi-threaded and dynamic language which give freedom to fasten the applications on any operating system. There is no denial to the fact that Java is the most important thing happened to IT industry across the globe after C++. It has an illustrious journey since its inception and has been the backbone of many new products and services that have surprised the world. Java is one of the most versatile computer programming languages ever. It is the best object oriented programming languages ever. So, it becomes even more important and equally curios to talk about the future of it.

For beginners, who are interested in learning Java, this book comprehensively covers all aspects of Java language and programming using Java. It was authored according to undergraduate University syllabus. Beginning with an introduction to Java language and its relationship with the internet and World Wide Web, it explores Java's object-oriented features, and then moves on to discuss advanced topics that are unique to Java. It is quite beneficial for those who want to learn the basics and fundamentals of Java Programming. This book contains a number of example programs. It has all that a reader needs to start programming in a right away. This book is for everyone who is either excited about internet or interested in Java programming. This book helps to enhance your programming skills and take yourself up the ladder of successful career.

Authors

SYLLABUS

Paper-III : OBJECT ORIENTED PROGRAMMING USING JAVA

Course Objectives

As the business environment becomes more sophisticated, the software development (software engineering is about managing complexity) is becoming increasingly complex. As of the best programming paradigm which helps to eliminate complexity of large projects, Object Oriented Programming (OOP) has become the predominant technique for writing software in the past decade. Many other important software development techniques are based upon the fundamental ideas captured by object oriented programming.

Course Outcomes

At the end of this course, student will:

1. Understand the concept and underlying principles of Object Oriented Programming
2. Understand how object oriented concepts are incorporated into the Java programming language
3. Develop problem-solving and programming skills using OOP concept
4. Understand the benefits of a well structured program
5. Develop the ability to solve real-world problems through software development in high-level programming language like Java
6. Develop efficient Java applets and applications using OOP concept
7. Become familiar with the fundamentals and acquire programming skills in the Java language.

UNIT - I

Fundamentals of Object Oriented Programming: Introduction, Object Oriented Paradigm, Basic Concepts of OOP, Benefits of OOP, Applications of OOP, Java Features: Overview of Java Language: Introduction, Simple Java Program Structure, Java Tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Command Line Arguments. Constants, Variables and Data Types: Introduction, Constants, Variables, Data Types, Declaration of Variables, Giving Value to Variables, Scope of Variables, Symbolic Constants, Typecasting, Getting Value of Variables, Standard Default Values; Operators and Expressions.

UNIT - II

Decision Making and Branching: Introduction, Decision making with if statement, simple if statement, if else statement, nesting of if else statements, the else if ladder, the switch statement, the conditional operator. Looping: Introduction, the while statement, the do-while statement, the for statement, jumps in loops.

Classes, Objects and Methods: Introduction, Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Constructors, Method Overloading, Static Members, Nesting of Methods.

UNIT - III

Inheritance: Extending a Class, Overloading Methods, Final Variables and Methods, Final Classes, Abstract Methods and Classes.

Arrays, Strings and Vectors: Arrays, One-dimensional Arrays, Creating an Array, Two-dimensional Arrays, Strings, Vectors, Wrapper Classes.

Interfaces: Multiple Inheritance: Introduction, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Assessing Interface Variables.

UNIT - IV

Multi-threaded Programming: Introduction, Creating Threads, Extending the Threads, Stopping and Blocking a Thread, Lifecycle of a Thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the ‘Runnable’ Interface.

Managing Errors and Exceptions: Types of Errors: Compile-time Errors, Run-time Errors, Exceptions, Exception Handling, Multiple Catch Statements, Using Finally Statement.

UNIT - V

Applet Programming: Local and Remote Applets, Applets and Applications, Building Applet Code, Applet Life Cycle: Initialization State, Running State, Idle or Stopped State, Dead State, Display State.

Packages: Introduction, Java API Packages, Using System Packages, Naming Conventions, Creating Packages, Accessing a Package, Using a Package.

Managing Input/Output Files in Java: Introduction, Concept of Streams, Stream Classes, Byte Stream Classes, Input Stream Classes, Output Stream Classes, Character Stream Classes: Reader Stream Classes, Writer Stream Classes, Using Streams, Reading and Writing Files.

Reference Books:

1. E. Balaguruswamy, Programming with Java: A Primer, 3e, Tata McGraw-Hill Company.
2. John R. Hubbard, Programming with Java, Second Edition, Schaum’s Outline Series, Tata McGraw-Hill Company.
3. Deitel and Deitel, Java TM: How to Program, PHI (2007).
4. D.S Mallik, Java Programming: From Problem Analysis to Program Design.
5. Object Oriented Programming through Java by P. Radha Krishna, University Press (2008).

Student Activity:

1. Create a front end using Java for the student database created
2. Learn the difference between ODBC and JDBC

OBJECT ORIENTED PROGRAMMING USING JAVA LAB

1. Write a program to perform various String Operations.
2. Write a program on class and object in Java.
3. Write a program to illustrate Function Overloading and Function Overriding methods in Java.

4. Write a program to illustrate the implementation of abstract class.
5. Write a program to implement Exception handling.
6. Write a program to create packages in Java.
7. Write a program on interface in Java.
8. Write a program to create Multiple Threads in Java.
9. Write a program to write Applets to draw the various polygons.
10. Write a program which illustrates the implementation of multiple inheritance using interfaces in Java.
11. Write a program to assign priorities to threads in Java.

CONTENTS

Unit I

Chapter 1

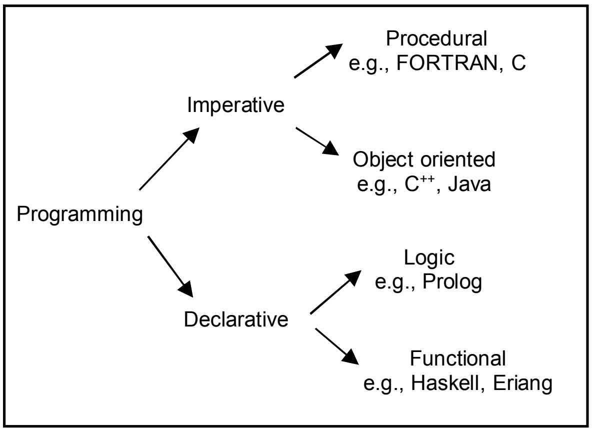
Fundamentals of Object Oriented Programming

PROGRAMMING IN JAVA

INTRODUCTION

The developments in software field continue to be dynamic. Various new tools and techniques are developed. This has made a compulsion on software industry and software engineers to always look for the new approaches or techniques for software design and development. The software design and development is becoming more and more critical in view of the increasing complexity of software systems as well as the highly competitive nature of the industry. There are two different types of programming approaches. They are:

1. Procedure oriented programming
2. Object oriented programming



PROCEDURE ORIENTED PROGRAMMING

A procedural program divides the code into smaller blocks called procedures. Procedure can interact with each other. In procedural programming, the program is written as a collection of actions that are carried out in a sequence, one after the other. The order is important.

For example: Prolog, Pascal, FORTRAN and PL/SQL

OBJECT ORIENTED PROGRAMMING (OOP)

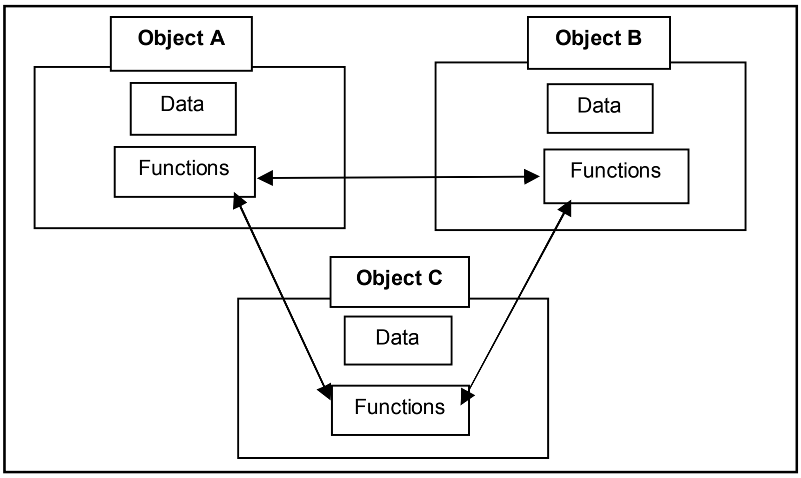
It is a revolutionary new way of writing computer programming. Earlier, computing involves mainly data and procedures that operate on data. In object oriented programming, construction is done by using objects. We use objects to interact with each other to send and receive messages. A program can have any number of objects and each object can be given a different task. The languages that support OOPs concept are:

For example: C++, Java and C#.

OBJECT ORIENTED PARADIGM

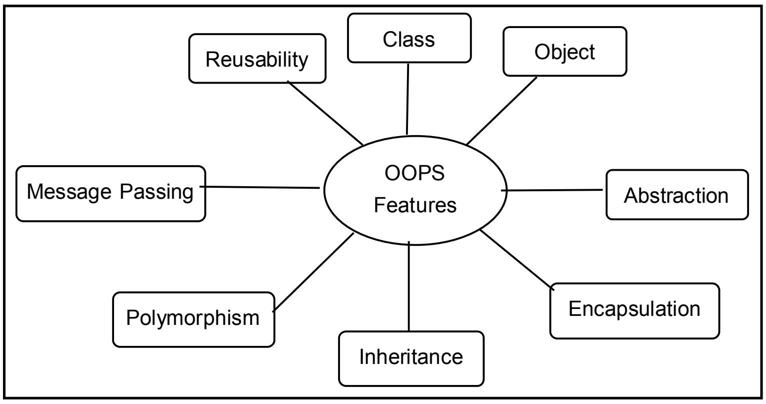
It is an approach to program organization and the development, which attempts to eliminate some of the failures of conventional programming methods by incorporating the best of structured programming features with several new concepts. Object oriented programming treats data as a critical element in the program development and does not allow freely around the system.

The object oriented programming provides a way of modularizing the programs by creating partitioned memory area for both data and functions. They can be used as templates for creating copies of such modules on demand. It allows us to decompose a problem into number of entities called objects and then builds data and functions around these entities. The organization of data and functions in object oriented programming are as shown in the figure given below.



BASIC CONCEPTS OF OOP

The general concepts of OOP which forms the heart of Java language are:

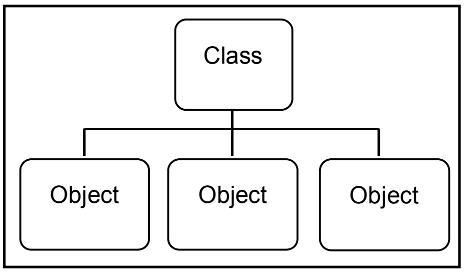


- Classes
- Objects
- Data Encapsulation
- Data Abstraction
- Inheritance
- Polymorphism
- Dynamic Binding
- Message Passing
- Reusability

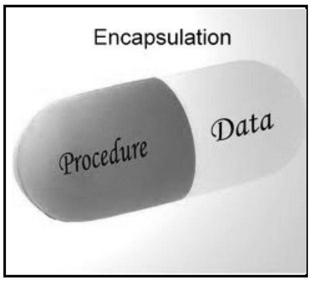
HPH

Classes: A class is a user-defined data type and behaves as a built-in data type. It is a collection of data and methods. Once a class has been defined, we can create any number of objects belonging to that class. An instance of class is called as object. A class and object are basic constructs of an object oriented programming language.

Objects: These are the basic runtime entities (units) in object oriented system. They may represent a person, a bank account, a table of data or any item that the program may handle. A problem in programming is analyzed in terms of objects and nature of communication between them. When a program is executed, objects interact with each other by sending messages.



Data Encapsulation: The wrapping (combining) up of data and methods into a single unit is called as encapsulation. The data is not accessible at outside of the world. Data is accessed by only those methods, which are wrapped in class. Data of one object cannot be accessible to other objects. The insulation of data from direct access by the program is known as data hiding.

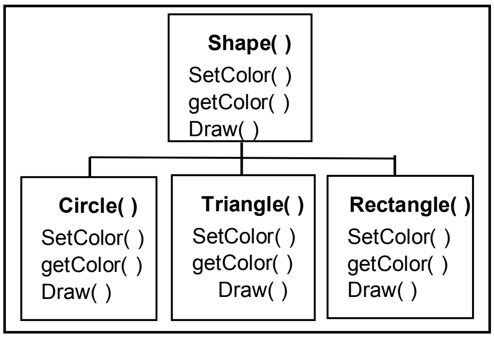


Data Abstraction: Abstraction refers to the act of representing essential features without including the background details or explanations. For example, when we apply brake to our two-wheeler, bike stops. But, we don't know the internal mechanism of how brake works. Still, we use break. Classes use the concept abstraction.

Inheritance: The process of deriving a new class from an existing class is known as inheritance. In this process, an object of one class acquires the properties of objects of another class. The new class is called derived class or subclass or child class and the existing class is called as base class or super class or parent class. It provides reusability, by adding additional features to an existing class without modifying it. There are five types of inheritances. They are.

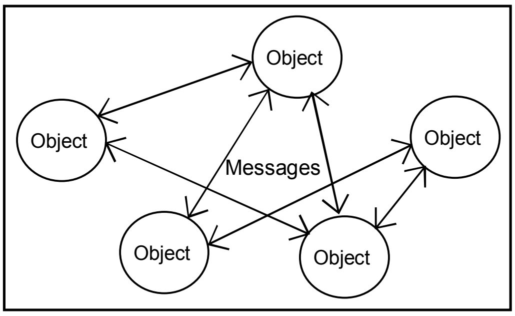
- Single inheritance
- Multiple inheritance
- Multi-level inheritance
- Hierarchical inheritance
- Hybrid inheritance

Polymorphism: Polymorphism means the ability to take more than one form. It is used extensively in implementation of inheritance. It plays an important role in allowing objects having different internal structure to share the same external interfaces. This operation exhibits different behavior in different times.



Dynamic Binding: Binding refers to the linking of a procedure call to the code to be executed in response to the call. Dynamic binding means that the code associated with a given procedure call is not known until the time of call at runtime.

Message Passing: Objects communicate with each other by sending and receiving information as people pass messages to one another. A message for an object is a request for execution of a procedure, and therefore will invoke a method in receiving the object that generates the desired result.



Interaction of objects via message passing

Reusability: The term reusability refers to the ability for multiple programs to use the same written and debugged existing class of data. This is time saving and adds code efficiency to the language. It optimizes code, helps in gaining secured applications and facilitates easier maintenance on the application.

Applications of OOP: The applications of OOP are beginning to gain importance in many areas. The most popular application of OOP has been in the area of user interface design such as Windows. There are hundreds of Window-based systems developed using OOP techniques. OOP is useful in this type of applications because it can simplify a complex problem. The main application areas of OOP are:

- User interface design such as Windows, menu…..
- Real-time systems
- Simulation and modeling
- Object oriented databases
- AI and Expert Systems
- Neural networks and Parallel programming
- Decision support and Office automation systems
- Hypertext, Hypermedia and Expert text
- CIM/CAM/CAD

OOP environment will enable the software industry to improve not only the quality of software systems but also its productivity.

Merits and Demerits of OOP

Benefits of OOP

OOP offers several benefits to the both program designer and the user. The new technology premises the greater productivity, better quality of software and lesser maintenance cost. The major benefits are:

- Through inheritance, we can eliminate redundant (repeated) code and extend the use of existing classes (reusability).
- We can build programs from standard working modules that communicate with one another. It leads to saving of development time and higher productivity.
- We can build secured programs using data hiding principle.

- It is possible to have multiple objects co-exist without any interface.
- It is possible to map objects in the problem domain to those objects in the program.
- It is easy to partition the work in a project based on objects.
- The data-centered design approach enables us to capture more details of a model in an implementable form.
- Message passing techniques for communication between objects make the interface descriptions with external systems much simpler.
- Polymorphism can be implemented, i.e., behavior of functions or operators or objects can be changed depending upon the operations.
- Easily upgraded from small to large systems.
- Software complexity can be easily managed.

Demerits of OOP

- It requires more data protection.
- Inadequate for concurrent problems
- Inability to work with existing systems.
- Compile time and runtime overhead.
- Unfamiliarity causes training overheads.

Difference between Procedure Oriented Programming and Object Oriented Programming

Procedure oriented programming	Object oriented programming
In POP, program is divided into small parts called functions.	In OOP, program is divided into parts called as objects.
It follows top-down approach.	It follows bottom-up approach.
Data can move freely from function to function.	Objects can move and communicate with each other through methods.
It is not easy to add new data and functions.	It is easy to add new data and functions.
Most of the functions share global data.	Every object can share global as well as local data.
It doesn't have any access specifiers.	It has access specifiers named public, private and protected.
It is less secure.	It is more secure.
It is very difficult to handle long and complex programs.	It is very simple to handle long and complex programs.
It does not have any way to hide the data.	It provides data hiding.
Data is in shareable mode.	Data is in non-shareable mode.
Overloading is not possible.	Overloading is possible.
e.g.: C, VB, Fortran, Pascal	***e.g.:*** C++, Java, VB.Net, C#.Net

FEATURES OF JAVA

The most striking features of Java made it as successful language. They made it as the first application language of World Wide Web. It is also a premier language for general purpose stand-alone applications.

- Compiled and Interpreted
- Platform Independent and Portable
- Object Oriented
- Robust and Secure
- Distributed
- Simple, Small and Familiar
- Multi-threaded and Interactive
- Dynamic and Extensible
- High Performance
- Ease of Development
- Scalability and Performance

Compiled and Interpreted: Generally, any computer language may be either compiled or interpreted. Java is both compiled and interpreted language. It involves two-stage system. Java compiler translates source code into byte code instructions. Byte code instructions are not machine instructions. Hence, Java interpreter generates machine code from byte code that can be directly executed by the machine.

Platform Independent and Portable: Java programs are portable. They can be easily moved from one system to another system anywhere and at anytime. It is not tied to any particular hardware or operating system. Hence, it is known as platform independent language.

Java achieves portability in two ways as:

1. Java compiler generates byte code instructions that can be implemented on any machine.
2. The size of primitive data types are machine independent.

Object Oriented: Java is purely object oriented language. Everything in Java is an object. The complete code and data reside within objects and classes.

Robust and Secure: Java is a robust language. It provides many safeguards in order to ensure reliable code. It has strict compile time and runtime checking for data types. It is designed as a garbage-collection language. It also has exception handling concept which captures errors and removes the risks. The absence of pointers ensures that the programs cannot access memory locations without authorization.

Distributed: Java was developed as a distributed language. It can be used for creating applications on networks. Java applications can open and can access remote objects on internet as easily as they can do in a local system. Java can be able to share both data and programs. This enables many programmers residing at different locations to work on a single project.

Simple, Small and Familiar: Java is not only simple but also a smaller language because it removes many features of C and C++ which can create problems. Java is familiar language because it is modeled on C and C++. Java is a simplified version of C++. Java does not use pointer, goto statement, operator overloading, multiple inheritance, etc.

Multi-threaded and Interactive: Java supports multi-threading. Multi-threading means handling multiple tasks simultaneously. Java supports multi-threaded programs. So, we need not wait for the application to finish one task before beginning another. The Java runtime comes with tools which support multiprocessor synchronization and construct interactive systems running smoothly.

Dynamic and Extensible: Java is a dynamic language. Java is capable of dynamically linking in new class libraries, methods and objects. Java also supports extensibility. Java program supports functions written in other languages such as C and C++. These functions are known as native methods which are linked dynamically at runtime.

High Performance: Because of using intermediate byte code, Java performance is excellent for an interpreted language. In order to reduce overhead during runtime, Java is designed carefully. Multi-threading improved overall execution speed of Java programs. Java is speed when compared with C and C++.

Ease of Development: Java supports several features of J2SE (Java 2 standard edition). These features reduce the work of programmer by reusability of code given by compiler. The source code give by compiler is free from bugs, so the program will be error-free.

Java Weaknesses:

Java is that it does not directly support true decimal data.

Example:

```
Double x = 5.02;
Double y = 0.01;
Double z = X + Y;
System.out.println(z);
```

Output:

```
5.029999999999999
```

But result is "5.03"

KEY TERMS AND CONCEPTS

Procedure oriented programming (POP)

Object oriented programming (OOP)

Class

Encapsulation

Data hiding

Inheritance

Reusability

Object

Method

Abstraction

Polymorphism

Message passing

Dynamic binding

SUMMARY

- There are two different types of programming approaches. They are:
  1. Procedure oriented programming
  2. Object oriented programming

- A procedural program divides the code into smaller blocks called procedures.
- The Object oriented programming provides a way of modularizing the programs by creating partitioned memory area for both data and functions that can be used as templates for creating copies of such modules on demand.
- An instance of class is called as object.
- A class is a collection of data and methods.
- The wrapping (combining) up of data and methods into a single unit is called as encapsulation.
- Abstraction refers to the act of representing essential features without including the background details or explanations.
- The process of deriving a new class from an existing class is known as inheritance.
- Polymorphism means the ability to take more than one form
- Binding refers to the linking of a procedure call to the code to be executed in response to the call.
- Objects communicate with each other by sending and receiving information as people pass messages to one another.

REVIEW QUESTIONS

1. Explain about the object oriented programming.
2. How are the data and methods organized in object oriented programming?
3. What are the basic concepts of object oriented programming?
4. What are the areas of application of object oriented programming technology?
5. What are the merits and demerits of object oriented programming?
6. What is the difference between procedure oriented and object oriented programming?
7. Explain about Java features.