

# Intro to OOP



Prepared by Harish Patnaik

**School of Computer Engineering, KIIT Deemed to be University**

# Content

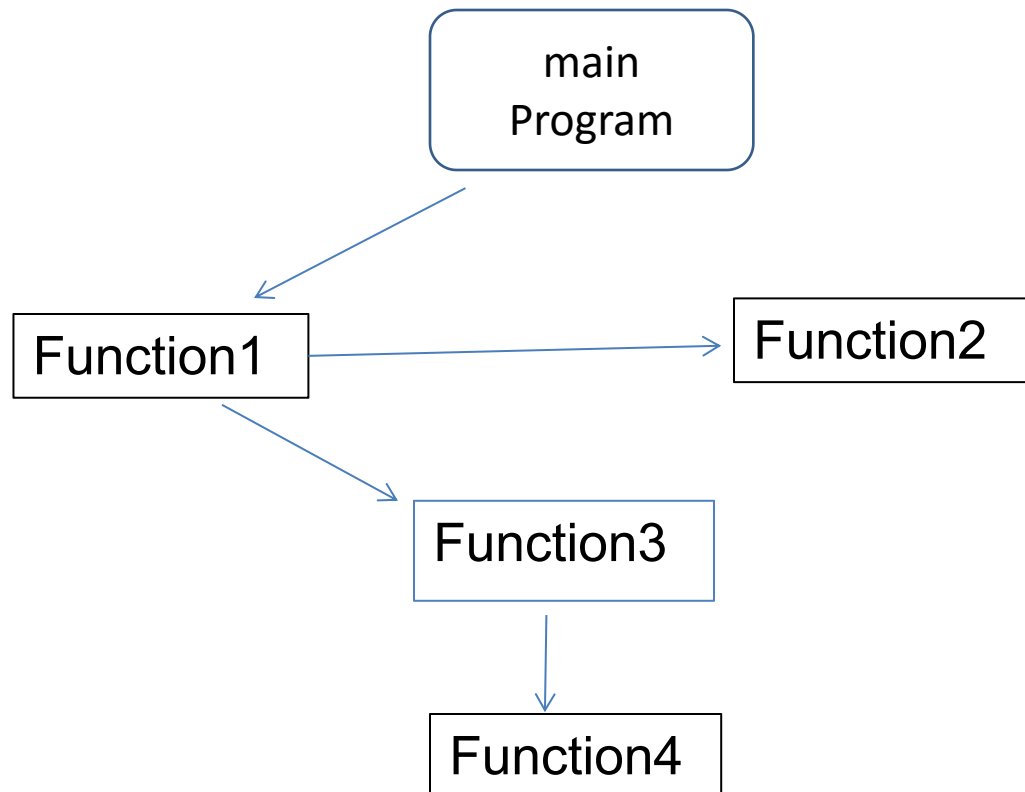
1. Introduction
2. Procedure Oriented Programming
3. Characteristics of OOP
4. Concepts of OOP

# Introduction

- It is another way of programming.
- It's objective is to provide clearer, reliable and easily maintainable approach to program design
- Different from Procedure Oriented Programming

# Procedure Oriented Programming

- Problem is viewed as a sequence of things to be done



# Characteristics of Procedure Oriented Programming

- Large programs are divided into smaller programs called Functions
- Emphasis on Functions
- Data is moved around the program in the form of local or shared data
- Function transform data from one form to another
- Program design approach is top-down

# Drawbacks of Procedural approach

- It does not model the real world problems very well.
- Most of the functions share global data
- Global data are more vulnerable to accidental change by a function
- The parts of the program are heavily dependent on each other

# Characteristics of Object Oriented Programming

- Programs are divided into set of objects
- Emphasis on data
- Data structures are designed to characterize the objects
- Data are hidden and cannot be accessed by external functions
- Objects may communicate through Functions
- Program design approach is bottom-up

# Concepts of Object Oriented Programming

## Object

- Objects are the basic runtime entities - customer, bank, account etc
- Programming problem is analyzed in terms of object and the nature of communication between them
- Program objects closely match with real world object
- It contains data and code to manipulate



# Concepts of Object Oriented Programming

## Class

- Class is a template for Objects
- Entire set of data and code of an object are defined in a class
- Any no. of objects can be created after declaration of class

Encapsulation - wrapping of data and function into a single unit (class)

- Data are accessible only by the function defined within the class - data hiding

# Concepts of Object Oriented Programming

## Abstraction -

- it refers to the act of representing essential features without including background details
- Class uses the concept of data abstraction and so they are called Abstract Data type

# Concepts of Object Oriented Programming

## Inheritance -

- it is the process by which object of one class acquires properties of object of another class.
- It facilitates reusability - one can add additional properties to an existing class without modifying it.

# Concepts of Object Oriented Programming

## Polymorphism -

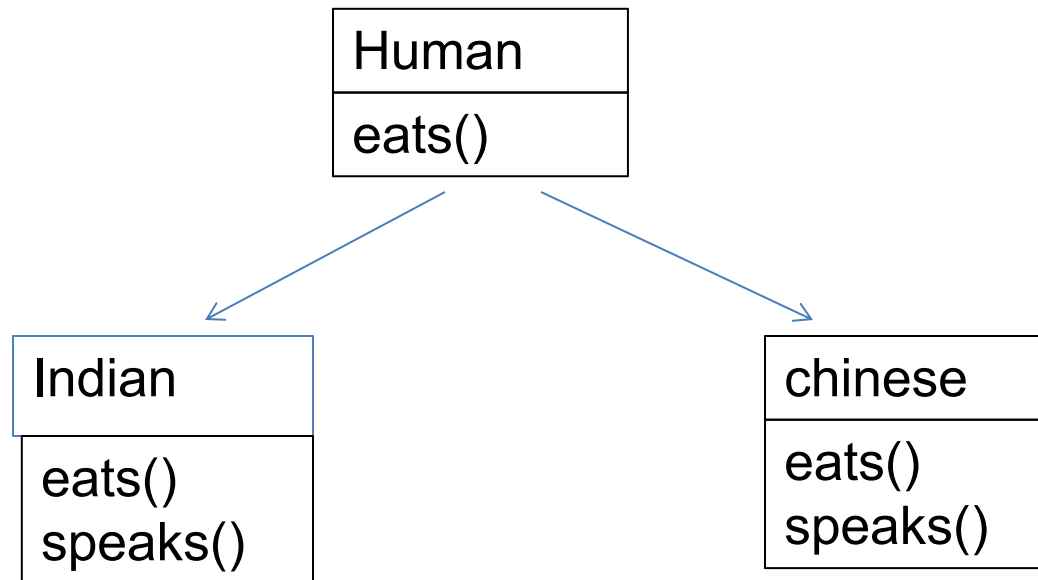
- poly means “many”, morphism means “form”
- An operation can take more than one form in different instances

Ex - add - two integers - addition operation  
- two strings - concatenation

- Polymorphism is extensively used in inheritance

# Polymorphism in inheritance

➤ Indian and Chinese inherits from Human



# Concepts of Object Oriented Programming

## Dynamic Binding -

- It is the linking of a function call to the function definition that will be executed on the call.
- Normally it is done during the linking process called *Early binding*.
- If the linking of a function call to the function definition is delayed until the run-time then is called *Late binding* (Dynamic binding).

# Concepts of Object Oriented Programming

## Message Passing -

- An object oriented program consists of set of objects that communicate with each other.
- Object communicate with one another by sending and receiving information.
- A message for an object is a request for execution of a procedure and therefore will invoke a function in the receiving object that generates the result.

# Concepts of Object Oriented Programming

## Benefits of OOP -

- It is possible to map objects of real problem to those of programs.
- Using inheritance one can extend the existing classes and eliminate the redundant codes.
- Data hiding principle helps to build secure programs.
- Multiple instances of a class can coexist.
- Such systems can be easily upgraded from small to large systems.
- Message passing techniques makes the interface description with external systems much simpler.
- Shorter development time



Thank you