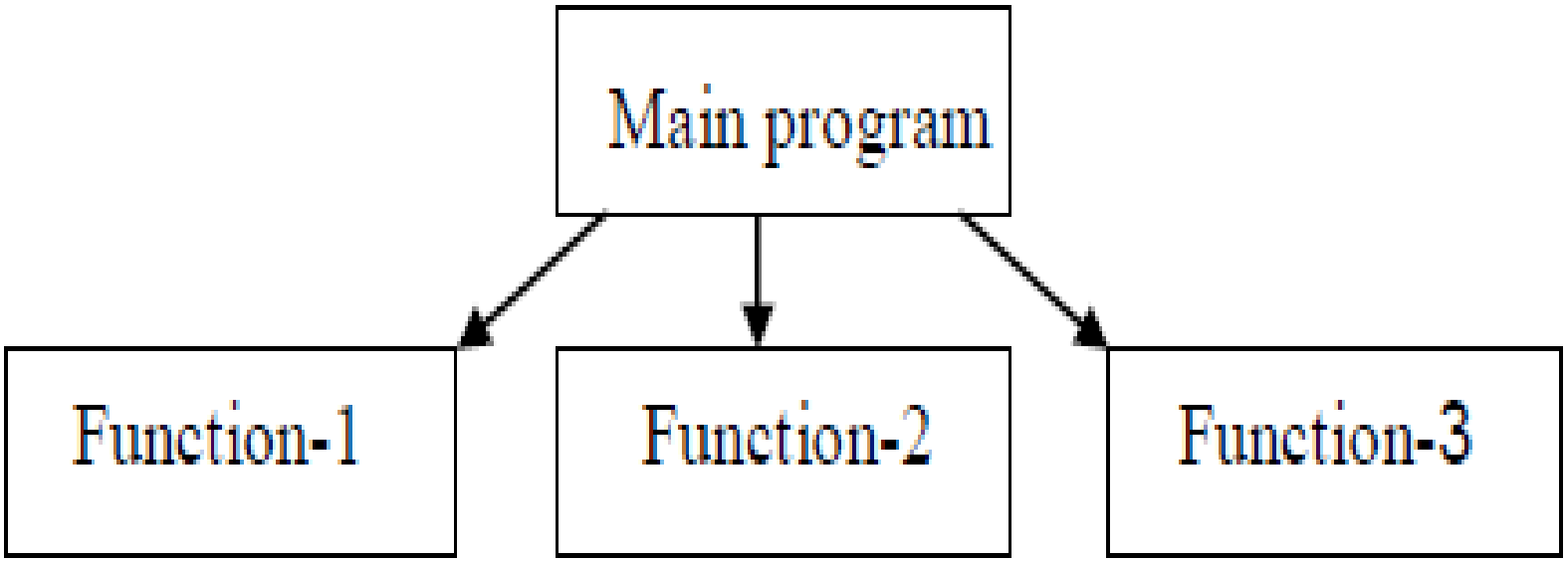
Slide 2

- Structured VS Object oriented Software Development

Characteristics of procedure oriented programming

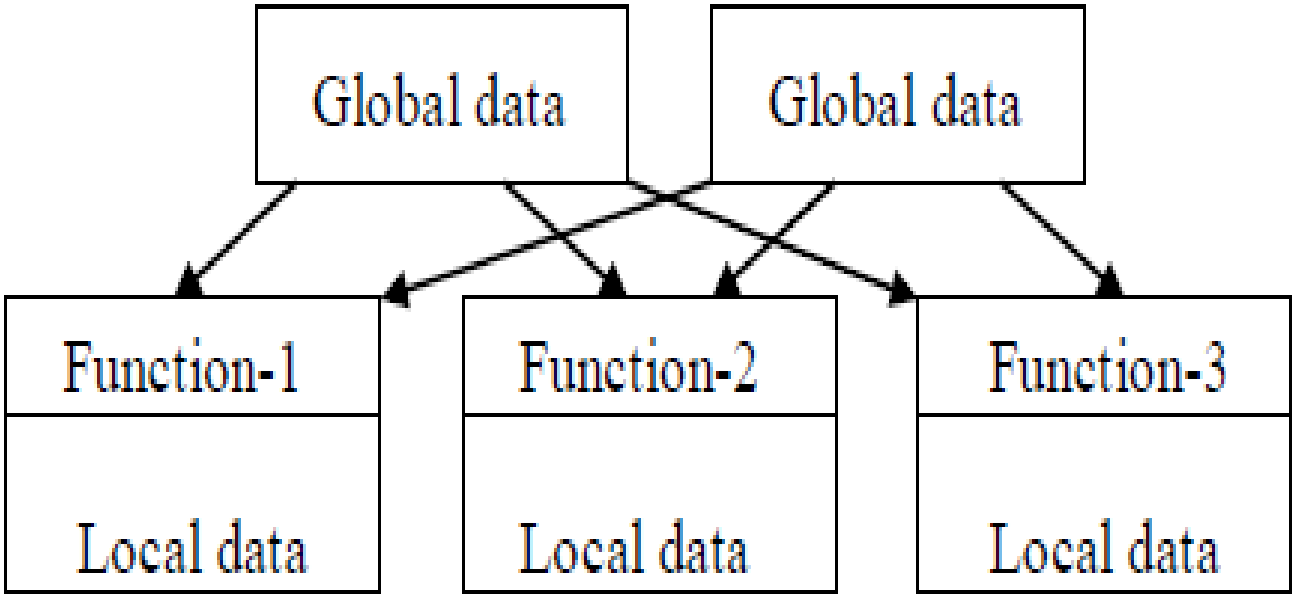
- Programs are divided into smaller programs known as functions.
- Most of the functions share global data
- Data move openly around the system from function to function
- Function transforms data from one form to another.
- Employs to Emphasis is on doing things (algorithm)
- Large p-down approach in program design

Procedure oriented programming basically consist of writing a list of instruction or actions for the computer to follow and organizing these instruction into groups known as functions.



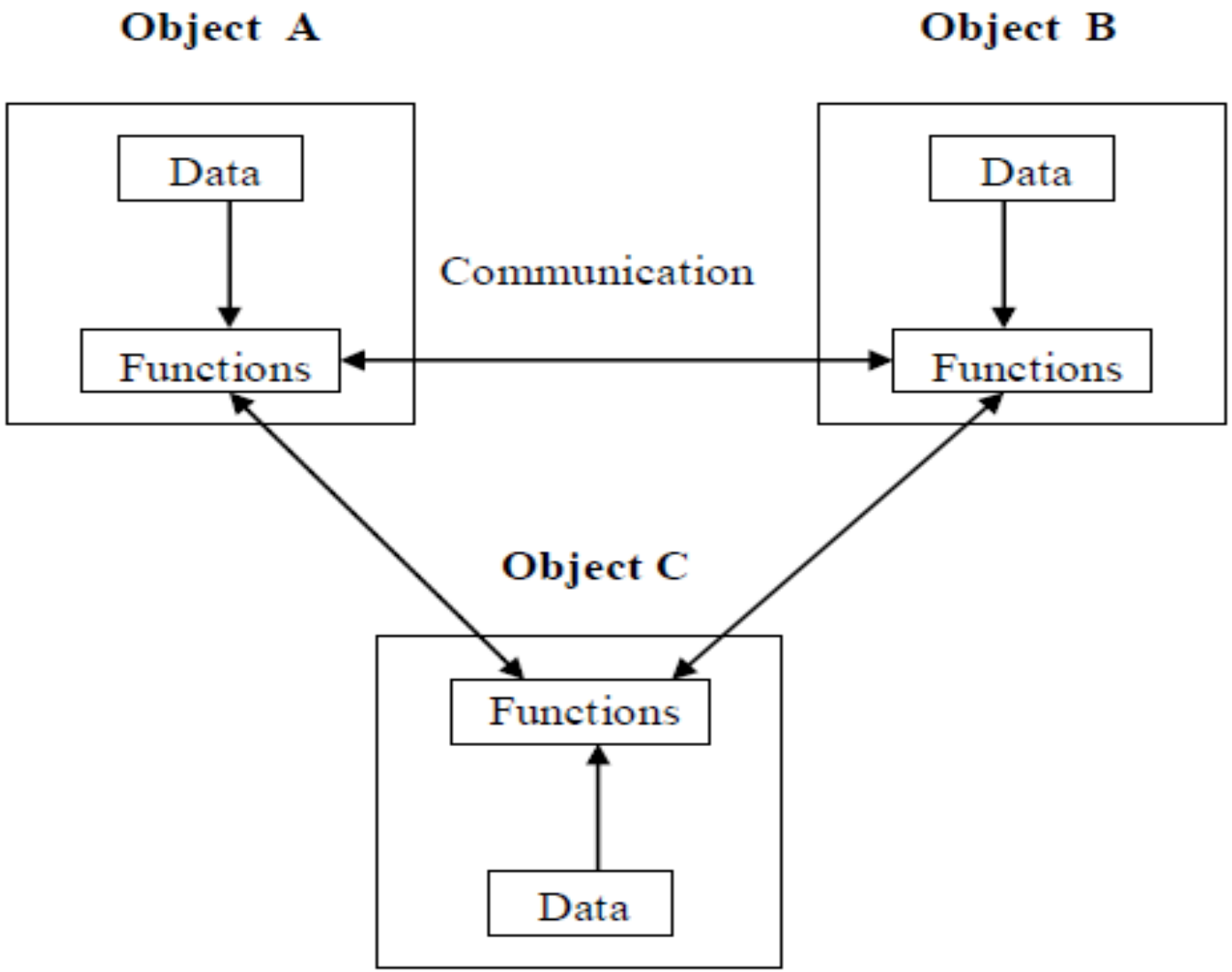
The disadvantage of the procedure oriented programming languages is:

1. Global data access
2. It does not model real word problem very well
3. No data hiding



Features of the Object Oriented programming

- Emphasis is on data rather than procedure.
- Programs are divided into what are known as objects.
- Data structures are designed such that they characterize the objects.
- Functions that operate on the data of an object are tied together in the data structure.
- Data is hidden and can't be accessed by external functions.
- Objects may communicate with each other through functions.
- New data and functions can be easily added.
- Follows bottom-up approach in program design.

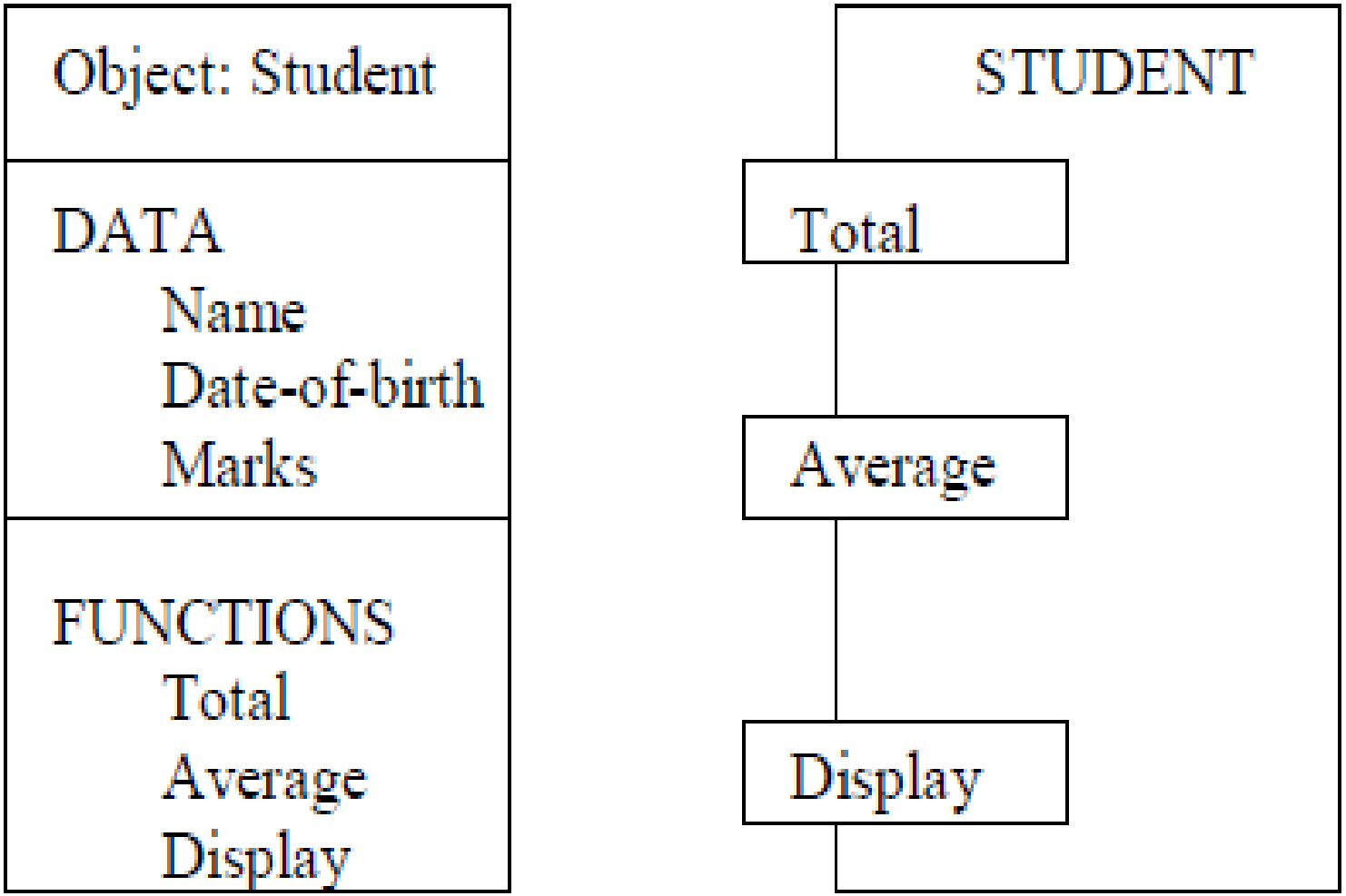

Object A
Object B
Data
Data
Communication
Functions
Functions
Object C
Functions
Data

OBJECTS ORIENTED CONCEPTS

- Objects
- Classes
- Data abstraction and encapsulation
- Inheritance
- Polymorphism
- Dynamic binding
- Message passing

OBJECTS

- Objects are the basic run-time entities in an object-oriented system. They may represent a person, a place, a bank account, a table of data or any item that the program must handle.
- The fundamental idea behind object oriented approach is to combine both data and function into a single unit and these units are called objects.
- The term objects means a combination of data and program that represent some real word entity.


Object: Student
DATA
Name
Date-of-birth
Marks
FUNCTIONS
Total
Average
Display
STUDENT
Total
Average
Display

CLASS:

- A class is a group of objects with similar properties and common behavior. A class represents a family of objects grouped together by virtue of their attribute.
- Classes are represented using class models or class diagrams which can be derived from object models
- A Class provides a template for the creation of objects

DATA ABSTRACTION

- Abstraction refers to the act of representing essential features without including the back ground details or explanations.
- Classes use the concept of abstraction and are defined using data members and functions to operate on the attributes.

DATA ENCAPSALATION

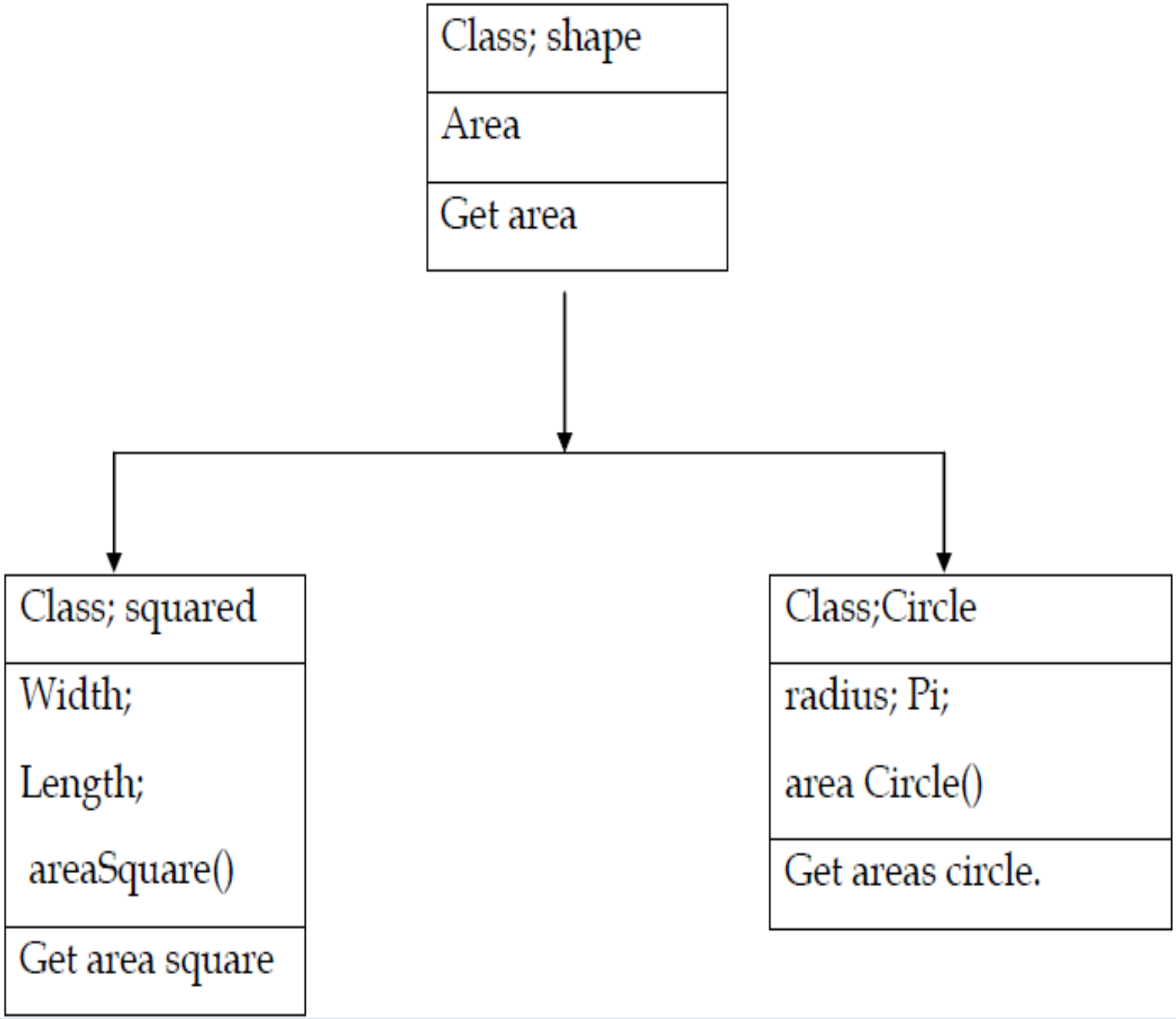
- The wrapping up of data and function into a single unit (called class) is known as encapsulation.
- The data is not accessible to the outside world and only those functions which are wrapped in the class can access it.
- These functions provide the interface between the objects data and the program.

INHERITENCE :

- Inheritance is the process by which objects of one class acquire the properties of another class.
- The concept of inheritance provides the idea of reusability. This mean that we can add additional features to an existing class with out modifying it. This is possible by designing a new class with the combined features of both the classes.
- The inheriting class called the sub class/child class/ derived class and the class from where the attribute are inherited is called the super class/base clase/ parent class.

POLYMORPHISIM

- Polymorphism is a Greek term that means "Having many forms or shapes". It refers to a situation where an object can exhibit different behavior
- Polymorphism means the ability to take more than one form. An operation may exhibit different instance. The behavior depends upon the type of data used in the operation.


Class; shape
Area
Get area
Class; squared
Width;
Length;
areaSquare()
Get area square
Class;Circle
radius; Pi;
area Circle()
Get areas circle.

MESSAGE PASSING

- An object oriented program consists of a set of objects that communicate with each other.
- A message for an object is a request for execution of a procedure and therefore will invoke a function (procedure) in the receiving object that generates the desired result.
- Message passing involves specifying the name of the object, the name of the function (message) and information to be sent.

Messages

- Messages in occur in the form of method invocations
- Syntax:
- obj.doSomething(arguments);
- e.g.
- person.birthday();

BINDING

- Binding refers to the linking of a procedure call to the code to the executed in response to the call.
- Static binding occurs during program compilation which Dynamic binding takes place during runtime.
- Dynamic binding means the code associated with a given procedure call is not known until the time of the call at run-time. It is associated with a polymorphic reference

Object-oriented analysis and design (OOAD)

- This is a software engineering approach that models a system as a group of interacting objects.
- Each object represents some entity of interest in the system being modeled, and is characterised by its class, its state (data elements), and its behavior.
- Various models can be created to show the static structure, dynamic behavior, and run-time deployment of these collaborating objects..

❖Object-oriented analysis (OOA) applies object-modeling techniques to analyze the functional requirements for a system.

❖Object-oriented design (OOD) elaborates the analysis models to produce implementation specifications. OOA focuses on *what the system does, OOD on how the system does it.*

Object-oriented systems

- An object-oriented system is composed of objects.
- The behavior of the system results from the collaboration of those objects.
- Collaboration between objects involves them sending messages to each other.

BENEFITS OF OO Software Development

- Object Oriented Technology offers several benefits to both the program designer and the user. Object-oriented contributes to the solution of many problems associated with the development and quality of software products. The principal advantages are :

❖ Through inheritance we can eliminate redundant code and extend the use of existing classes.

❖W e can build programs from the standard working modules that communicate with one another, rather than having to start writing the code from scratch. This leads to saving of development time and higher productivity.

❖ The principle of data hiding helps the programmer to build secure programs that can’t be invaded by code in other parts of the program.

❖ It is possible to have multiple instances of an object to co-exist with out any interference.
❖ It is easy to partition the work in a project based on objects.
❖ Object-oriented systems can be easily upgraded from small to large systems.
❖ Message passing techniques for communication between objects makes the interface description with external systems much simpler.
❖ Software complexity can be easily managed.

METHODOLOGY,METHOD, MODEL

- The process of building computer software and information systems has been always dictated by different development methodologies.
- A software development methodology refers to the framework that is used to plan, manage, and control the process of developing an information system

❖A methodology consist of phases and sub phases that guide the system developer in the choice of technique that are appropriate for each stage of the IS project. This includes support services such as resource allocation, scheduling and reporting standards.

What is a method?

❖A method refers to a particular way that can be used in performing a task. It provides a step by step procedure or description of doing a job

What is a model?

A model is a method express in graphical nature. A model represents a process or logic. A model is a representation of a reality. System development rely heavily on model. This is because of the following reasons.

- ❖They represents something that may not be easily represented in text
- ❖Model acts as visual aid
- ❖They represent the essential features only
- ❖The real may not be available
- ❖Picture is worth a thousand words (a model simplifier what it represents)
- ❖Model help users to gain more insight into the system

Tools and techniques

The choice of a methodology will influence the choice of technique and tools. A technique is a specific way of dealing with a stage [activity]in system development

A tool is a support for a method (activities)

Most tools are graphical in nature (models) e.g. in analysis

- ❖ Tools (DFDs, ERMs), use cases class diagram
- ❖ Design decision tables , Trees
- ❖ Testing CAST (Computer Aided Software Testing)

Exercise

- Discuss Structured System Analysis and Design Methodology (SSADM)

THE END.

WAIRAGU G.R