

ACKNOWLEDGMENT

- Publish material by Virtual University of Pakistan.
- Publish material by Deitel & Deitel.
- Publish material by Robert Lafore.

WHAT YOU HAVE DONE SO FAR !

- Introduction to Computer Programming (ITC):
 - How to think a program.
 - How to write a program.
 - Basic Programming structure.
 - Procedural paradigm.
 - Group of functions that interact with each other.
- Already have knowledge about the offered course.
- Need to strengthen our concepts.
- Try to implement what we know.

COMPUTER PROGRAMMING AS A COURSE

What we will study:

- Object Oriented Programming.
- How to think in a OOP way.
- How to map real world into a program
- Or, how to program a real world scenario.
- Aim :
 - Our aim is to learn the concepts of Object Oriented programming.
 - Try to digest them.
 - Implement in a program.
 - Tool: C++.

CONTENTS OF THE COURSE

- Object Oriented Programming.
- Classes & Objects.
- Overloading.
- Inheritance.
- Polymorphism.
- Generic Programming.
- Exception Handling.

BOOKS

Text Book:

1- C++ How to program By Deitel & Deitel.

Reference Books:

1- The C++ Programming Language By Bjarne Stroustrup.

2- Object Oriented Software Engineering By Jacobson.

GRADING POLICY

Quiz's	10	at least five
Assignments	10	at least three
Project	10	one in group
Midterm's	30	two (15 marks each)
Final	40	
<hr/>		
Total	100	

WEEK ONE, CLASS ONE

- Introduction to the Generation of Languages
- Introduction to Programming Paradigms

WHAT ARE THE TYPES OF PROGRAMMING LANGUAGES

First Generation Languages

Second Generation Languages

Third Generation Languages

Fourth Generation Languages

Fifth Generation Languages

FIRST GENERATION LANGUAGES

Machine language

- **Operation code** – such as addition or subtraction.
- **Operands** – that identify the data to be processed.
- Machine language is machine dependent as it is the only language the computer can understand.
- Very efficient code but very difficult to write.

SECOND GENERATION LANGUAGES

Assembly languages

- Symbolic operation codes replaced binary operation codes.
- Assembly language programs needed to be “assembled” for execution by the computer. Each assembly language instruction is translated into one machine language instruction.
- Very efficient code and easier to write.

THIRD GENERATION LANGUAGES

High Level Languages

Closer to English but included simple mathematical notation.

- Programs written in **source code** which must be translated into machine language programs called **object code**.
- The translation of source code to object code is accomplished by a machine language system program called a **compiler**.

THIRD GENERATION LANGUAGES (CONT'D.)

Alternative to compilation is interpretation which is accomplished by a system program called an [interpreter](#).

Common third generation languages

- FORTRAN
- COBOL
- C and C++
- Visual Basic

FOURTH GENERATION LANGUAGES

Event based programming

A high level language (4GL) that requires fewer instructions to accomplish a task than a third generation language.

Used with databases

- Query languages
- Report generators
- Forms designers
- Application generators



FIFTH GENERATION LANGUAGES

Artificial Intelligence

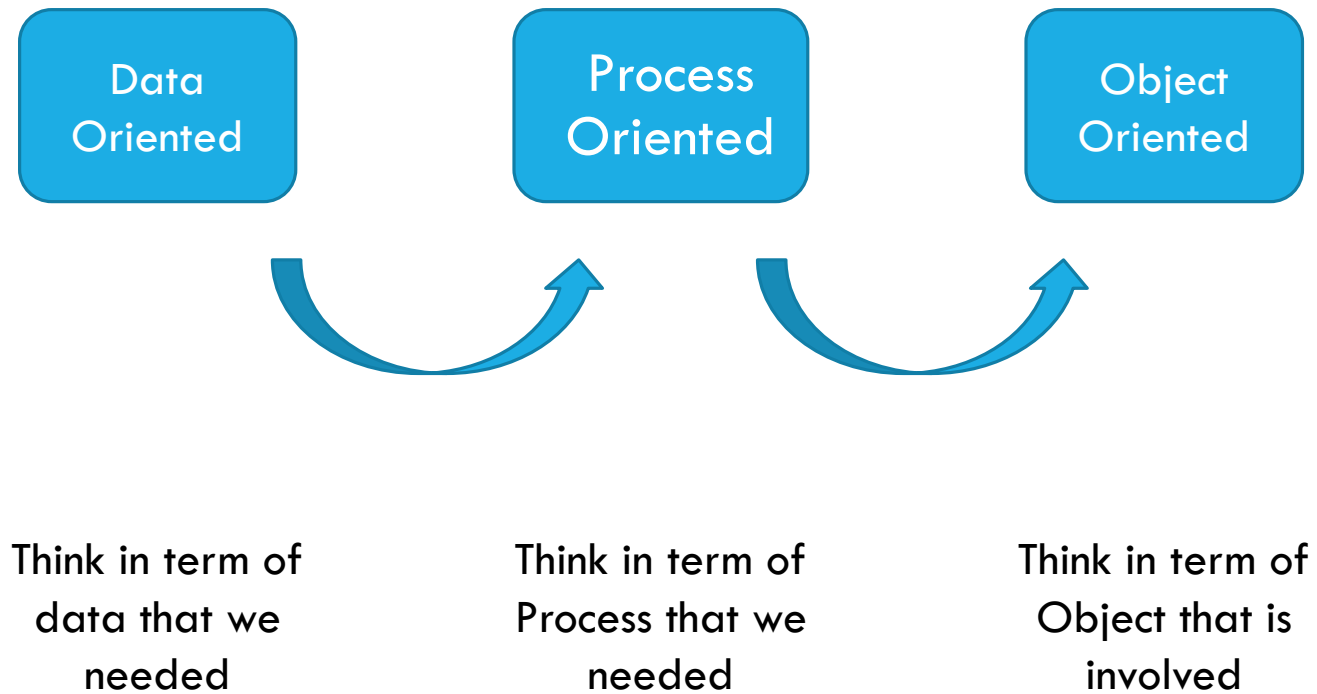
Machines that can think like humans

Machines that takes its own decisions

PROGRAMMING PARADIGMS

- 1- Sequential
- 2- Procedural
- 3- Object Oriented

PROGRAMMING PARADIGMS



WEEK ONE, CLASS TWO

- Introduction to OO paradigm
- Principles of Object Oriented Paradigm

WHAT IS OBJECT ORIENTATION

- A technique for system modeling.
- OO model consists of several interacting objects.

WHAT IS A MODEL?

- abstraction of something.
- Purpose is to understand the product before developing it.

EXAMPLE — OO MODEL



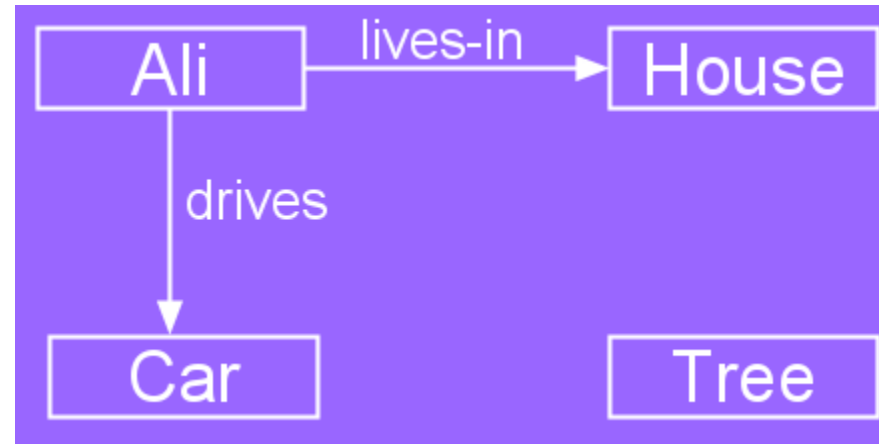
EXAMPLE — OO MODEL

Objects

- Person i.e Name: Ali
- House
- Car
- Tree

Interactions

- Ali lives in the house
- Ali drives the car



OBJECT-ORIENTATION - ADVANTAGES

- People think in terms of objects
- OO models map to reality

Therefore, OO models are:

- easy to develop
- easy to understand

FIVE PRINCIPLES OF OO PARADIGM

- 1- Abstraction: To have the relevant information.
- 2- Encapsulation: To hide information inside the object.
- 3- Polymorphism: To have many shapes / behaviors.
- 4- Inheritance: To create a new object with an existing one (To adopt features from others)
- 5- Reusability: Ability to use an object again and again if needed.

WHAT IS AN OBJECT?

An object is:

- It can be anything for which we want to save Information
- Something tangible (Ali, Car)
- Something that can be captured intellectually (Time, date)

An object has:

- State / attributes / properties / data
- Well-defined behavior / methods / functions
- Unique identity

ALI AS AN OBJECT

Attributes:

- Name
- age

Behavior (operations)

- Walks
- Eats

Identity

- His name

CAR AS AN OBJECT

State (attributes)

- Color
- Model

Behavior (operations)

- Accelerate
- Start Car
- Change Gear

Identity

- Its registration number