**What is Object-Oriented?**

**We will look at few practical examples of systems implemented using OOD.**

**Example 1.**

**A digital watch.**

**A B**

**12:15:00**

**Objects :**

* **Watch Display**
* **Buttons A(set mode), B(adjust time)**
* **Timer**

**Example 2.**

**Traffic light controller**

**Objects:**

* **Traffic light**
* **Lamps**
* **Sequencer**
* **Traffic sensor**
* **Priority scheduler**

**Example 3.**

**GUI for Image Processing**

**Objects:**

* **Image**

**-Binary**

**- Greylevel**

**- Colour**

**- Stereo**

**- Video**

**- Volume**

* **Sensor**

**- Camera**

**- Laser range finder**

* **Algorithms**
* **Display**

**- Monochrome**

**- 8-bit pseudo-colour**

**- 24-bit full-colour**

**Example 4.**

**Automated teller machine (ATM) - system to support a banking network of ATMs.**

**Central Computer**

**ATM**

**ATM**

**ATM**

**Bank Computer**

**Bank Computer**

**Account**

**Account**

**Account**

**Account**

**Objects:**

* **ATM**
* **Account**
* **Transaction**
* **Bank**
* **Customer**
* **Consortium**

**What is Object-Oriented?**

**3 components:**

* **Data abstraction**

**- implementation, interface**

* **Encapsulation**

**- ‘packaged’ objects**

**- controlled interaction with other objects**

* **Polymorphism (‘many forms’)**
  + - * 1. **- objects can assume ‘many forms’ through**
        2. **same interface**

**Why Object-Oriented?**

**Allows the management of complexity.**

**How?**

**Controlled object behaviour and interactions as defined by the object *state.***

**Also**

**Allows for extendibility and re-usability.**

**How?**

**Classes are packaged with well defined interfaces and hidden implementations.**

**What is OOD good for?**

**Modelling asynchronously interacting objects**

* **GUIs**
* **Event simulation**
* **Ray tracing visualisation**
* **CAD simulation**
* **Real-time control/embedded systems**
* **Robotics**
* **Image/Video processing**
* **Client/Server systems**

**What is it not good for?**

* **Mathematical programming**
* **Data processing/data manipulation**

**Why C++?**

**Supports OOP :**

* **Data abstraction**

**Classes**

* **Encapsulation**
* **Polymorphism Inheritance, Virtual functions**