

## 5.A. Programación orientada a objetos.

### 2. Conceptos de la orientación a objetos.

Como hemos visto, la orientación a objetos trata de resolver un problema simulando los elementos que participan en su resolución y basa su desarrollo en los siguientes conceptos:

- **Abstracción.** Permite capturar las características y comportamientos similares de un conjunto de objetos con el objetivo de darles una descripción formal. La abstracción es clave en el proceso de análisis y diseño orientado a objetos, ya que mediante ella podemos llegar a definir un conjunto de clases que permitan modelar la realidad, o el problema que se quiere atacar.
- **Encapsulación.** Organiza los datos y métodos de una clase, evitando el acceso a datos por cualquier otro medio distinto a los definidos. El estado de los objetos sólo debería poder ser modificado desde métodos de la propia clase.
- **Modularidad.** Propiedad que permite subdividir una aplicación en partes más pequeñas (llamadas módulos), cada una de las cuales debe ser tan independiente como sea posible de la aplicación en sí y de las restantes partes. En orientación a objetos es algo consustancial, ya que los objetos se pueden considerar los módulos más básicos del sistema.
- **Principio de ocultación.** La implementación de una clase sólo es conocida por los responsables de su desarrollo. Gracias a la ocultación, ésta podrá ser modificada para mejorar su algoritmo de implementación sin tener repercusión en el resto del programa.
- **Polimorfismo.** Consiste en reunir bajo el mismo nombre comportamientos diferentes. El modo en el que actuará un objeto en cada momento depende del mensaje que se le envíe al solicitar la ejecución de sus métodos.
- **Herencia.** Relación que se establece entre objetos en los que unos utilizan las propiedades y comportamientos de otros formando una jerarquía. Los objetos heredan las propiedades y el comportamiento de todas las clases a las que pertenecen.

- **Recolección de basura.** Técnica por la cual el programa se encarga de destruir automáticamente los objetos que hayan quedado sin referenciar, y por tanto desvincular su memoria asociada. Característica disponible en algunos lenguajes orientados a objetos, no en todos.