

### ***1.1 ¿En la programación orientada a objetos qué es el constructor y qué función cumple?***

El constructor es un método especial de una clase que se ejecuta automáticamente cuando se crea un objeto. Su función es inicializar los atributos del objeto y dejarlo listo para usarse. No devuelve ningún valor y generalmente tiene el mismo nombre que la clase.

### ***1.2 ¿En la programación orientada a objetos qué es un objeto y qué función cumple?***

Un objeto es una instancia de una clase. Representa una entidad con propiedades (atributos) y comportamientos (métodos). Su función es permitir modelar elementos del mundo real dentro del programa y manipular datos de manera estructurada.

### ***1.3 ¿En la programación orientada a objetos qué es una clase y qué función cumple?***

Una clase es una plantilla o modelo a partir de la cual se crean los objetos. Define los atributos y métodos que tendrán esos objetos. Su función es organizar el código y agrupar datos y funciones relacionadas en una sola estructura.

### ***1.4 ¿Qué diferencias destacadas se aprecian entre la OOP y la programación estructurada (secuencial)?***

Programación Orientada a Objetos: - Se basa en objetos y clases. - Alta reutilización gracias a la herencia y el encapsulamiento. - Los datos están ligados a los objetos. - Mantenimiento más fácil y modular. Programación Estructurada: - Se basa en funciones y procedimientos. - Reutilización limitada. - Los datos y las funciones están separados. - Más difícil de mantener en programas grandes.

### ***1.5 ¿Para qué y por qué se usan modificadores de acceso?***

Los modificadores de acceso se usan para controlar la visibilidad y el acceso a los atributos y métodos de una clase. Permiten proteger la información y evitar que partes externas del programa modifiquen datos internos sin control. Ejemplos comunes son: public (accesible desde cualquier parte), private (solo dentro de la clase) y protected (accesible desde la clase y sus subclases).