

Paradigma	Características	Ventajas	Desventajas
Orientado a objetos	<ul style="list-style-type: none">- Está pensado para tener una trazabilidad muy directa con el mundo real- Utiliza objetos y no algoritmos como bloques fundamentales de código- Los objetos tienen comportamientos y características, y colaboran entre sí- Cada objeto representa una instancia de una clase y cada clase está relacionada con otra a través de distintos tipos de relaciones- Abstracción: Permite seleccionar características relevantes del problema y descartar aquellas que no son fundamentales. De esta manera podemos identificar comportamientos comunes y así definir nuevas clasesEncapsulamiento: Permite reunir a todos los elementos que pueden considerarse pertenecientes a una misma entidadOcultación: Cada objeto está aislado del exterior y tiene una interfaz que define como puede interactuar con otros objetos. Esto ayuda a proteger la información interna y evitar que se modifique desde el exteriorCohesión: Mide las responsabilidades asignadas a cada objetoAcoplamiento: Mide las relaciones entre los objetos	<ul style="list-style-type: none">- Es una forma mas natural de modelar los problemas- Permite manejar de forma más sencilla la complejidad.- Facilita el mantenimiento y la extensión de los lenguajes- Fomenta el reúso de los componentes mejorando la productividad- Permite la relación entre distintos conjuntos de datos- Permite aislar más fácilmente los posibles errores que puedan surgir- Permite reducir la cantidad de código redundante	<ul style="list-style-type: none">- El rendimiento de ejecución se reduce- No se recomienda para tareas pequeñas- Cambia la forma de pensar la programación tradicional por lo que la curva de aprendizaje es más amplia- Los objetos al ser abstractos no suele coincidir la visión de un programador con la de otro
Estructurado	<ul style="list-style-type: none">- Esta pensado en términos de separación de los problemas en datos por un lado y comportamiento por otro asemejándose más a las computadoras- Se limita a procesar las instrucciones en el orden en que se han escrito, es decir, que tiene una ejecución lineal o secuencial- Tiende a ser orientado a la acción. La unidad fundamental de la programación es la acción.	<ul style="list-style-type: none">- Se descomponen los problemas en piezas más pequeñas lo que simplifica el problema- Tiene una estructura sencilla de entender por lo que la curva de aprendizaje es menor- La ejecución tiene mayor rendimiento- Es útil para programas de pequeña envergadura	<ul style="list-style-type: none">- No permite manejar la complejidad de forma sencilla dado que todo el programa es un solo bloque de código- Al efectuar un cambio es difícil controlar su impacto en el resto del sistema-Si el sistema es muy grande puede resultar problemático la detección de fallas- La forma de modelar los problemas está muy orientada a como lo hace la computadora y no tanto a como es en la realidad