

Antipatrones de programación



- Packratting: Consumir recursos de memoria en exceso debido a no liberar objetos reservados dinámicamente una vez que ya han dejado de ser utilizados.
- El Dios: Un programa omnipresente y desconocido. Aquel sistema donde una sola clase ó modulo (la función main o equivalente) hace todo.
- Lava Flow: construir grandes cantidades de código de manera desordenada, con poca documentación y poca claridad de su función en el sistema.
- Fantasmas: Demasiadas clases en un programa o tablas en una base de datos. Varias clases o tablas con mínimas responsabilidades.
- Reinventar la rueda: Se refiere a reimplementar componentes que se pueden conseguir prefabricados de antemano, y hacer poco reuso en el código.
- Input kludge: No especificar e implementar el manejo de entradas inválidas.
- Optimización temprana: cuando invertimos una cantidad considerable de tiempo intentando mejorar el rendimiento de nuestro código, antes de saber siquiera si funciona.
- Código ravioli: Construir sistemas con multitud de objetos muy débilmente conectados.
- Ocultación de errores: Capturar un error antes de que se muestre al usuario, y reemplazarlo por un mensaje sin importancia o ningún mensaje en absoluto.

Personalmente creo que El Dios es el más común empezando a programar debido al escaso conocimiento sobre las clases, métodos, objetos...
Haciendo que los programadores novatos (entre los cuales evidentemente me sitúo) caigan en este problema junto con el Lava Flow probablemente.