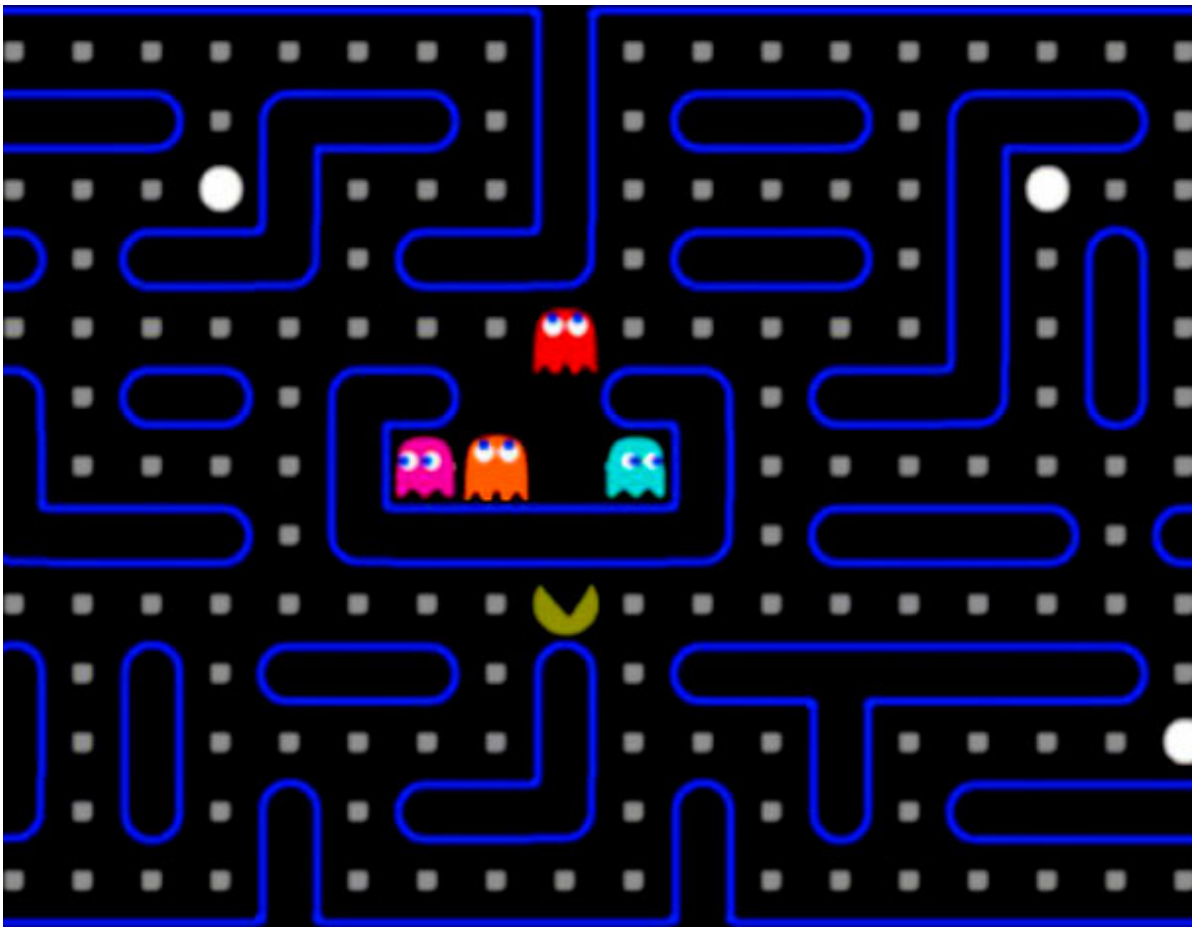
Videojuegos: retos y perspectivas de la programación en el siglo XXI



Pacman. FOTOS: Internet.

Colaboración Especial

Por Julio Enrique Salceda Estrada

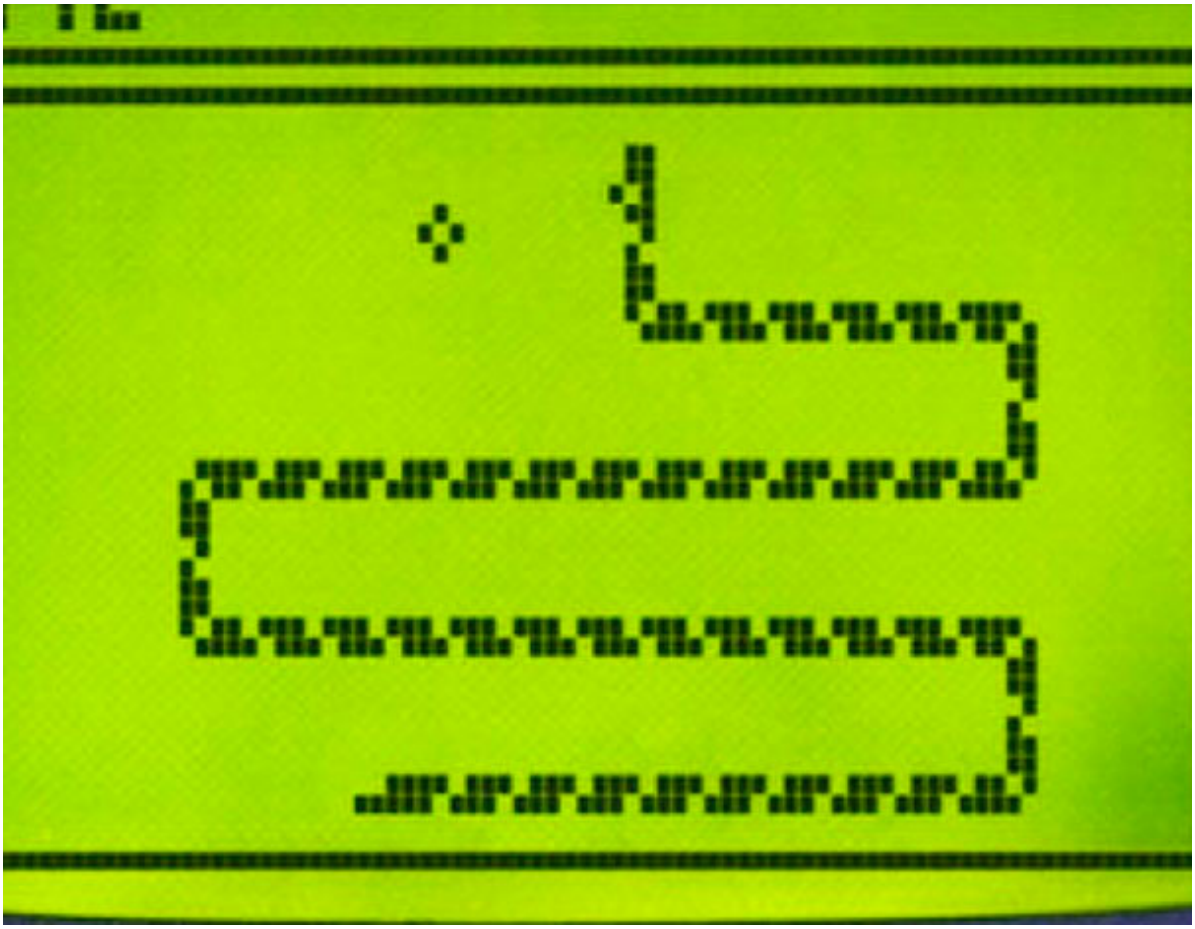
La Paz, Baja California Sur (BCS). Haciendo un repaso rápido, los juegos como ***Pacman, Tetris*** y ***Pin Pong*** se han hecho clásicos a través de los años. Y no podemos dejar de lado al

juego de ***Snake***, que va comiendo y aumentando de tamaño hasta que uno tropieza con su cola o se anuncia el fin del juego. ¿Quién diría que juegos así serían referencias para la **cultura *milenial***, y fueran ampliamente aceptados? ¿Qué hace que unos **juegos virtuales** que fueron buenos en su momento ya no lo sean hoy?

Veamos algunas características que nos pueden acercar a la tan anhelada *Valhala* de los programadores de **videojuegos**. Nótese que esta lista no es exhaustiva y tampoco es una receta para el juego perfecto. Sin embargo, los buenos juegos comparten una o varias de estas características.

El tema visual

Los programadores suponemos que la jugabilidad y la propaganda de un **videojuego** es todo lo que se necesita para tener éxito, pero no aplica a todos los videojuegos. Con tema visual me refiero a los usos del color, su brillo y contraste. Puede un juego estar en escala de grises e incluso ser popular, pues tenemos su imagen en nuestras mentes. Imaginen un *pacman* que no sea amarillo, o un *tetris* sin sus múltiples bloques de colores, y sería monótono. Pero si los jugamos en blanco y negro, o escala de grises, en nuestras mentes son coloridos y divertidos.



El Snake, que se jugaba desde un celular.

El pase de nivel

Para que un juego sea atractivo debe suponer un reto que vaya de la mano con la destreza del jugador. El programador supone que debe ir poniendo un nivel tras otro, aumentando la dificultad y eso enganchará al jugador. Lo que realmente se necesita es que el jugador vea diferente el primer nivel cada vez que lo juega, y que el que se vuelva más difícil de vencer sea el jugador, no el **videojuego.** Un error de cálculo en el nivel de aprendizaje puede hacer que el jugador se frustre innecesariamente y deje de jugar, o sea tan fácil que pierda el interés. Así que lo mejor es que nos dediquemos a programar un juego que sea bueno y dejemos al jugador la responsabilidad de crecer o no con él. Solo preocupémonos en que no sea frustrante y se agradecerá.

Las vidas

Un programador de **videojuegos** debe entender que el jugador

tiene un proceso de inmersión en el juego. El no ganar o perder no es una opción. Las respuestas deben ser obvias, pero no confusas. La experiencia del jugador debe ser enfocado a su destreza, nosotros –los **programadores de videojuegos**– a la inteligencia. El cómo cae un objeto suavemente o estrepitosamente dependiendo de su peso o elasticidad, el cómo brinca o se apoya en la pared es nuestra responsabilidad, no del jugador. Incluso cosas como caer en un hoyo y reaparecer cerca –si nos quedan vidas– son cosas que al jugador no deben preocupar pero nosotros debemos tomar en cuenta.

El final

Un prototipo de juego quizá no contemple el final pero es algo que debe estar en la mente del programador. Es una responsabilidad del programador de **videojuegos** dar un final satisfactorio. Uno o múltiples finales han sido memorables para cada jugador, y probablemente no recuerden de qué se trató el juego pero sí de cómo fue el final. Debe ser una proeza memorable.

¿Te ha parecido útil este artículo? No olvides dejar tu mensaje en la caja de comentarios.