

M1 Programación I FIN A

ACTIVIDAD 1

|  |  |
| --- | --- |
| **Tutor:** | **Adriana Hernández García** |
| **Estudiante:** | **José Ramón Ibáñez Posadas** |
| **Matricula:** | **BNL098377** |

|  |  |
| --- | --- |
| Monterrey, Nuevo León | domingo, 14 de Enero de 2024 |

INTRODUCCIÓN

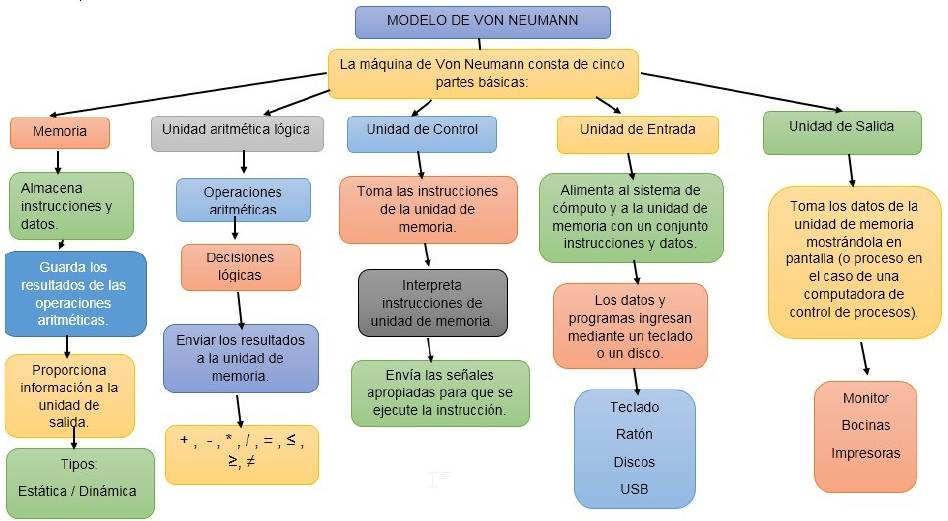
El modelo de von Neumann, propuesto por el matemático y físico John von Neumann a mediados del siglo XX, ha sido fundamental en el desarrollo de las computadoras modernas y ha sentado las bases para comprender cómo se ejecutan los programas. Este modelo, caracterizado por su arquitectura de almacenamiento único para datos e instrucciones, ha influido de manera significativa en la evolución de los lenguajes de programación. Los lenguajes de programación son la interfaz esencial entre los humanos y las computadoras, permitiendo la creación de software mediante la expresión de instrucciones que las máquinas pueden comprender y ejecutar. En este contexto, exploraremos las características clave de los lenguajes de programación, sus diversos tipos, y las fases cruciales involucradas en la elaboración de programas.

.

DESARROLLO

MAPA CONCEPTUAL

# **MODELO DE VON NEUMANN**



RESUMEN

# **LENGUAJES DE PROGRAMACIÓN**

Los lenguajes de programación son sistemas formales utilizados para comunicarse con una computadora. Sirven como intermediarios entre los humanos y las máquinas, permitiendo la creación de software mediante la especificación de instrucciones que la computadora puede entender y ejecutar. Estos lenguajes facilitan la creación de programas informáticos al traducir las instrucciones escritas por los programadores a un formato que la máquina puede procesar.

# **CARACTERÍSTICAS DE LOS LENGUAJES DE PROGRAMACIÓN**

1. **Sintaxis y Semántica:** Cada lenguaje tiene reglas específicas de sintaxis y semántica que determinan cómo se deben escribir las instrucciones y cómo serán interpretadas por la máquina.
2. **Portabilidad:** Algunos lenguajes son más portátiles que otros, lo que significa que un programa escrito en ese lenguaje puede ejecutarse en diferentes plataformas sin necesidad de modificaciones significativas.
3. **Eficiencia:** La eficiencia del código, en términos de uso de recursos como memoria y procesamiento, varía entre lenguajes de programación.
4. **Facilidad de Aprendizaje:** La facilidad con la que los programadores pueden aprender y utilizar un lenguaje es una característica importante.
5. **Orientación a Objetos, Funcionalidad, o Procedimental:** Los lenguajes pueden ser orientados a objetos, funcionales, o procedimentales, lo que influye en la forma en que se organiza y estructura el código.

# **TIPOS DE LENGUAJES DE PROGRAMACIÓN**

1. **Lenguajes de Bajo Nivel:** Incluyen ensamblador y máquina, proporcionando un mayor control sobre el hardware pero siendo menos legibles y más difíciles de aprender.
2. **Lenguajes de Alto Nivel:** Más cercanos al lenguaje humano, facilitan la programación y la legibilidad del código. Ejemplos incluyen Python, Java y C++.
3. **Lenguajes de Scripting:** Diseñados para la automatización de tareas. Ejemplos son JavaScript, Perl y Python.
4. **Lenguajes de Programación Funcional:** Se centran en la programación funcional y evitan el estado mutable. Ejemplos son Haskell y Lisp.
5. **Lenguajes de Programación Orientados a Objetos:** Se basan en conceptos de objetos y clases. Ejemplos incluyen Java y C#.

# **FASES DE LA ELABORACIÓN DE UN PROGRAMA**

1. **Análisis:** Comprender los requisitos del software y definir qué debe hacer el programa.
2. **Diseño:** Planificar la estructura del programa, seleccionar algoritmos y decidir sobre la estructura de datos.
3. **Codificación:** Escribir el código fuente del programa en un lenguaje de programación específico.
4. **Pruebas:** Verificar que el programa funcione según lo previsto, corrigiendo errores y asegurándose de que cumple con los requisitos.
5. **Implementación:** Introducir el software en el entorno de producción.
6. **Mantenimiento:** Realizar ajustes y correcciones a medida que sea necesario, respondiendo a cambios en los requisitos o resolviendo problemas descubiertos después de la implementación.

CONCLUSIÓN

En conclusión, el modelo de von Neumann ha dejado una huella indeleble en la informática al establecer los principios fundamentales que rigen la ejecución de programas. A su vez, los lenguajes de programación han evolucionado para ofrecer a los desarrolladores herramientas cada vez más poderosas y expresivas. La elección del lenguaje, guiada por sus características y el contexto del proyecto, desempeña un papel crucial en el éxito de la programación. Las fases para la elaboración de un programa, desde el análisis hasta el mantenimiento, proporcionan un marco sistemático para el desarrollo de software, asegurando la calidad y la adaptabilidad de los programas en un mundo tecnológico en constante cambio. En este equilibrio entre arquitectura de computadoras, lenguajes de programación y procesos de desarrollo, se forja la base de la innovación y la capacidad de la humanidad para dar forma y aprovechar la revolución digital.

BIBLIOGRAFÍA

