

Sistemas y Computación

Systems and Computing

Autor: **Santiago Suaza Builes**

IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia

Correo-e: **s.suaza@utp.edu.co**

Resumen— Este documento presenta un resumen de los principales contenidos del programa de Ingeniería de Sistemas y Computación. En el documento se explica el sentido de las cuatro grandes temáticas que se abordan en la carrera, y se indican sus principales aplicaciones en el campo industrial e investigativo. Las áreas son: programación, redes y comunicaciones, ingeniería de software e inteligencia artificial. El docente ha realizado la primera parte: programación, dejando para el estudiante la realización de los restantes tres temas: redes, software e inteligencia artificial.

Palabras clave— sistemas, redes, inteligencia artificial, software, computación, investigación, industria.

Abstract— This document presents a summary of the main contents of the Computer and Systems Engineering program. The document explains the meaning of the four major themes that are addressed in the career, and indicates their main applications in the industrial and research field. The areas are: programming, networks and communications, software engineering and artificial intelligence. The teacher has done the first part: programming, leaving the student to carry out the remaining three topics: networks, software and artificial intelligence.

Key Word— systems, networks, artificial intelligence, software, computing, research, industry.

I. INTRODUCCIÓN

El Programa Ingeniería de Sistemas y Computación estudia varios campos del conocimiento ligados a la teoría de la Informática y los Sistemas en general. Se han identificado varias áreas que representan el sustento teórico y práctico de la carrera, según se ha mencionado en el resumen del documento.

El objetivo del presente documento es describir cada uno de los temas mencionados, buscando con ello brindar una visión integral de la carrera, lo cual le permitirá al estudiante elegir aquellas temáticas que mejor se adapten a sus capacidades académicas.

PROGRAMACIÓN

En [1] se define la programación de la siguiente manera: “La programación informática es el proceso por medio del cual se diseña, codifica, limpia y protege el código fuente de programas computacionales. A través de la programación se dictan los pasos a seguir para la creación del código fuente de programas informáticos. De acuerdo con ellos el código se escribe, se prueba y se perfecciona.”

Si se analiza la anterior definición, se aprecia que la programación se orienta a la solución de problemas técnicos y cotidianos a través de la escritura de un cierto código fuente, el cual debe respetar cierta estructura y método de trabajo. Para programar se debe conocer, con un buen grado de detalle, un lenguaje que se adapte al problema que se desea resolver.

Por ejemplo, si el problema a resolver es de carácter matemático, lo usual es que se emplee un lenguaje como Python, de gran acogida en los últimos tiempos. Una variante, más antigua pero igualmente importante, es el lenguaje Fortran, con el cual se desarrollaron las primeras soluciones a los problemas de Ingeniería.

Si el problema es de tipo comercial, un lenguaje que se utilizó ampliamente es el lenguaje COBOL. Se dice que en la actualidad, y por un factor histórico, el 80% de las soluciones informáticas comerciales están elaboradas con este lenguaje.

Si la idea es resolver un problema de tipo general, se puede recurrir al lenguaje C, el cual se puede considerar como el padre de todos los lenguajes, pues fue utilizado en los orígenes de la computación moderna para el desarrollo del primer sistema operativo importante: UNIX.

Los lenguajes de programación se organizan según su modelo y estructura. A cada una de estas formas de organización se la conoce como: “Paradigma de Programación”.

Según [2] un paradigma de programación es:

“Un paradigma de programación es un marco conceptual, un conjunto de ideas que describe una forma de entender la construcción de programa, como tal define:

- Las herramientas conceptuales que se pueden utilizar para construir un programa (objetos, relaciones, funciones, instrucciones).
- Las formas válidas de combinarlas.

Los distintos lenguajes de programación proveen implantaciones para las herramientas conceptuales descritas por los paradigmas. Existen lenguajes que se concentran en las ideas de un único paradigma así como hay otros que permiten la combinación de ideas provenientes de distintos paradigmas.”.

Existen muchos paradigmas de programación. Los más importantes se describen a continuación:

PARADIGMA ESTRUCTURADO

El paradigma estructurado se basa en la ejecución secuencial y ordenada de instrucciones sobre un espacio de memoria debidamente organizada. Las estructuras básicas de programación son: secuencia, decisión y ciclo. Un lenguaje clásico de la programación estructurada es el lenguaje C.

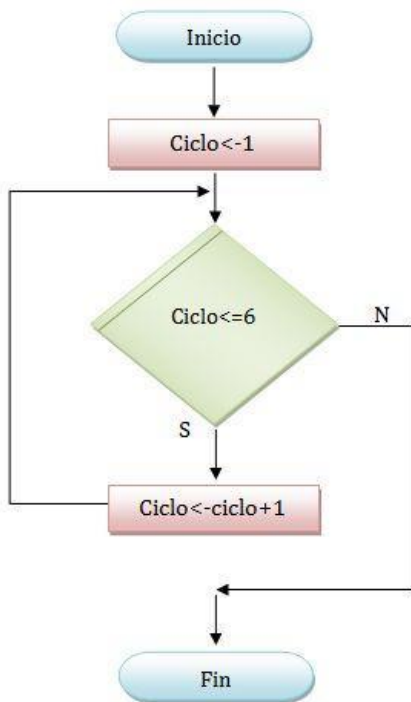


Figura 1. Paradigma estructurado

PARADIGMA DE OBJETOS

El paradigma de objetos es una concepción en la cual definen entidades, denominadas clases, a partir de las cuales se crean objetos que interactúan entre sí. En cierto sentido, el paradigma de objetos es similar al concepto de objeto que se percibe en el mundo que nos rodea. Un lenguaje orientado a objetos es Smalltalk.

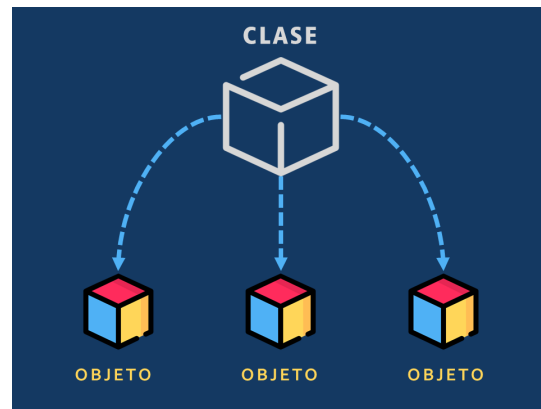


Figura 2. Paradigma orientado a objetos

PARADIGMA LÓGICO

El paradigma lógico está basado en la lógica de predicados de primer orden. Su objetivo es permitir extraer conclusiones a partir de premisas, de acuerdo con un conjunto de reglas y mecanismos de inferencia. Un lenguaje en el campo de la lógica es el PROLOG.

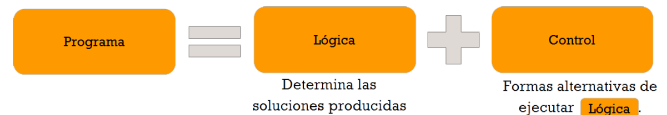


Figura 3. Paradigma lógico

PARADIGMA FUNCIONAL

El paradigma funcional se basa en la utilización de funciones como base de relación entre las partes de un programa. Una función es una porción de código que cumple un objetivo específico, permitiendo con ello simplificar y automatizar las tareas. Un lenguaje funcional es HASKELL.

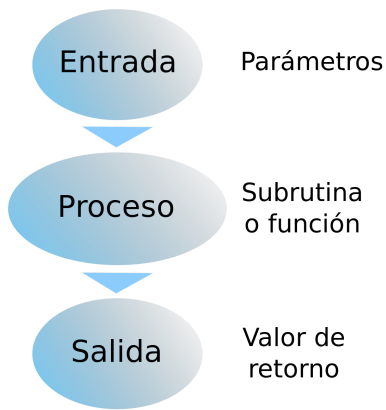


Figura 4. Paradigma funcional.

El paradigma estructurado se conoce, en ciertos entornos, como el paradigma IMPERATIVO. En la siguiente gráfica se aprecia lo visto hasta el momento:



Figura 5. Paradigmas de programación

Los paradigmas de programación, a su vez, se organizan en dos grandes categorías. La primera de ellas se conoce con el nombre de categoría IMPERATIVA. La segunda es la categoría DECLARATIVA.

La diferencia entre las dos categorías es la siguiente: en la categoría IMPERATIVA, los lenguajes de programación requieren que se indique de manera minuciosa cada uno de los pasos de la solución del problema. En este modelo se requiere realizar un seguimiento secuencial de cada paso a resolver en tal modelo.

En la categoría DECLARATIVA los lenguajes de programación no requieren de una descripción detallada y minuciosa de cada paso de la solución. Los lenguajes de tipo declarativo se caracterizan por disponer de un motor interno que les permite simplificar la ejecución de un programa. El motor le permite a los lenguajes encontrar caminos de solución que no están disponibles en el modelo imperativo.

En la siguiente gráfica se aprecia dicha clasificación.



Figura 6. Lenguajes imperativos y declarativos

Por último, se presenta un gráfico que presenta los principales lenguajes de programación.

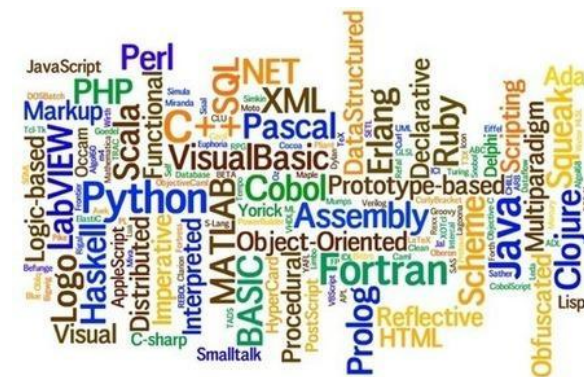


Figura 7. Lenguajes de programación.

REDES Y COMUNICACIONES

COMMUNICATION & NETWORKING

Antes todos los sistemas de computación eran altamente centralizados. Debido al rápido progreso tecnológico la forma como se recolecta, transporta, almacena y procesa la información ha cambiado. La unión entre computadores y comunicaciones tiene una profunda influencia en la forma como los sistemas computacionales están organizados. El antiguo modelo de un sólo computador sirviendo a todas las necesidades computacionales de la organización ha sido reemplazado por otro en el cual un gran número de computadores independientes, pero interconectados hacen el trabajo. Estos sistemas son llamados redes de computadores.

Otra forma de definir redes de computadores: es un sistema formado por dos, tres, cuatro o más computadores interconectados entre sí a través de una tarjeta de Red. Las señales de Red se suelen transmitir, por medio de cables de Red, que son los que van conectados directamente a la tarjeta de red de la computadora o se puede realizar la conexión de forma inalámbrica.

Objetivos de una Red de Computadores

- Compartir recursos: Hacer que todo el Software y Hardware esté disponible para cualquiera en la red.
- Aumentar la confiabilidad: Conectar fuentes alternativas de suministro. Tener respaldo para los diferentes servicios de la red.
- Ahorro económico
 - Relación precio/desempeño
 - Varios microprocesadores con un servidor.
 - Modularidad
- Medio de comunicación para gente distante: Con el empleo de la red, comunicarse es relativamente fácil.
- Ofrecer servicios a los usuarios.



1.3 Servicios de una red de computadoras

- ✓ Impresión: Permite compartir impresoras entre varios ordenadores de la red.
- ✓ Información: Los servidores de información pueden almacenar bases de datos para su consulta por los usuarios de

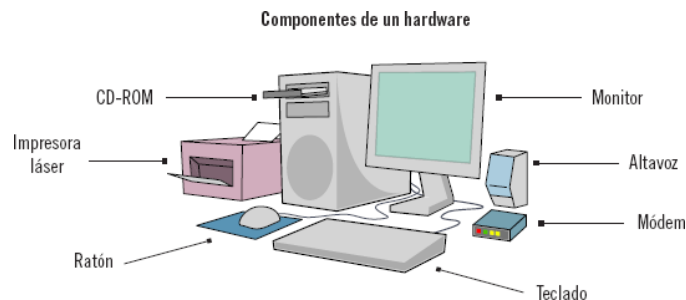
la red u otro tipo de información, como por ejemplo documentos de hipertexto. (WEB) .

- ✓ Comunicación: Permiten la comunicación entre los usuarios a través de mensajes escritos. (EMAIL) .
- ✓ Telefonía: Permite la comunicación voz sobre IP.
- ✓ Otros

1.4 Partes esenciales de una red de computadoras

✓ Hardware:

- Cables de Red : Son los que van conectados a la tarjeta de Red, actúa como medio de transmisión por donde pasan las señales entre los equipos. Los tipos son Cable Coaxial, Cable de par Trenzado y Cable de Fibra Óptica.
- Tarjetas de Red : Estas nos permiten conectarnos a una Red (NIC Network Interface Card – Tarjeta de Interfaz de Red).
- Periféricos compartidos : Son los dispositivos que van conectados a la red por medio de un cable de red UTP, que es también llamado cable de par trenzado con extensión RJ45 ó también se pueden conectar de forma inalámbrica.



✓ Software:

- Sistema Operativo de Red : Administra y gestiona las comunicaciones y recursos de la red como programas, los datos y aplicaciones entre otros.
- Driver o Controladora de Red : Los controladores de Red, hacen que la tarjeta de Red pueda ser reconocida por el sistema operativo.
- Programas de aplicación en Red : Son aquellos programas con los cuales navegamos o usamos en internet.



- La ingeniería de software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable, que sea fiable y trabaje en máquinas reales (Bauer, 1972).

- La ingeniería de software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación, y mantenimiento del software. Standard Glossary of Software Engineering Terminology

INGENIERÍA DE SOFTWARE

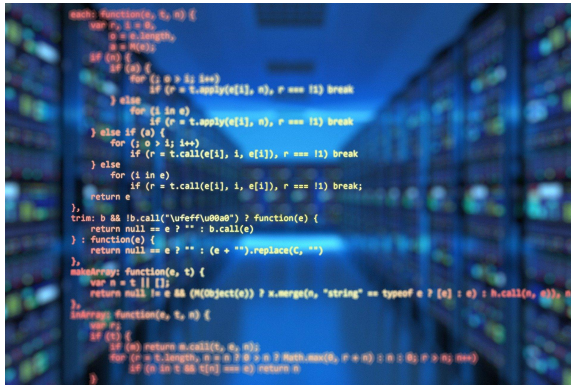
La Ingeniería de *Software* es una de las ramas de las ciencias de la computación que estudia la creación de software confiable y de calidad, basándose en métodos y técnicas de ingeniería. Brindando soporte operacional y de mantenimiento, el campo de estudio de la ingeniería de *software*. Integra ciencias de la computación, ciencias aplicadas y las ciencias básicas en las cuales se encuentra apoyada la ingeniería.

Se citan las definiciones más reconocidas, formuladas por los siguientes prestigiosos autores:

- Ingeniería de software es el estudio de los principios y metodologías para el desarrollo y mantenimiento de sistemas software (Zelkovitz, 1978).
- Ingeniería de software es la aplicación práctica del conocimiento científico al diseño y construcción de programas de computadora y a la documentación asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como desarrollo de software o producción de software (Bohem, 1976).

objetivos

- Mejorar el diseño de aplicaciones o *software* de tal modo que se adapten de mejor manera a las necesidades de las organizaciones o finalidades para las cuales fueron creadas.
- Promover mayor calidad al desarrollar aplicaciones complejas.
- Brindar mayor exactitud en los costos de proyectos y tiempo de desarrollo de los mismos.
- Aumentar la eficiencia de los sistemas al introducir procesos que permitan medir mediante normas específicas, la calidad del *software* desarrollado, buscando siempre la mejor calidad posible según las necesidades y resultados que se quieren generar.
- Una mejor organización de equipos de trabajo, en el área de desarrollo y mantenimiento de *software*.
- Detectar a través de pruebas, posibles mejoras para un mejor funcionamiento del *software* desarrollado.



INTELIGENCIA ARTIFICIAL

En términos sencillos, inteligencia artificial (IA) se refiere a los sistemas o las máquinas que imitan la inteligencia humana para realizar tareas y que tienen la capacidad de mejorar iterativamente a partir de la información que recopilan. La IA se manifiesta de varias formas. Algunos ejemplos son:

- Los bots conversacionales que utilizan IA para comprender más rápido los problemas de los clientes y proporcionar respuestas más eficientes
- Los asistentes inteligentes utilizan la IA para analizar información crítica proveniente de grandes conjuntos de datos de texto libre para mejorar la programación
- Los motores de recomendación pueden proporcionar recomendaciones automatizadas para programas de TV según los hábitos de visualización de los usuarios

La IA se trata mucho más sobre el proceso y la capacidad de pensamiento superpoderado y el análisis de datos que sobre cualquier formato o función en particular. Aunque la IA muestra imágenes de robots de aspecto humano de alto funcionamiento que se apoderan del mundo, la IA no pretende reemplazar a los humanos. Su objetivo es mejorar

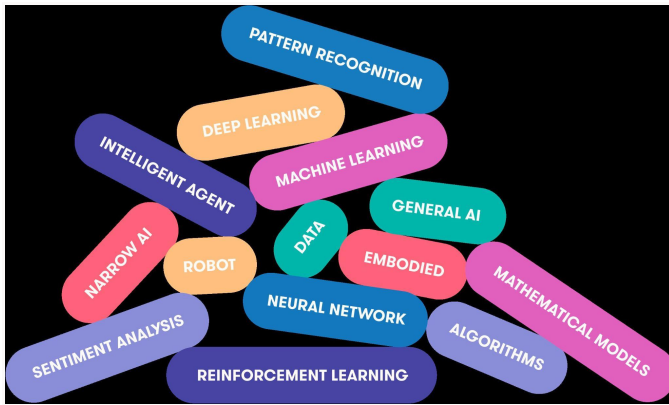
significativamente las capacidades y contribuciones humanas. Eso la convierte en un activo comercial muy valioso.



Términos de inteligencia artificial

La IA se ha convertido en un término general para las aplicaciones que realizan tareas complejas que antes requerían aportes humanos, como la comunicación en línea con los clientes o jugar al ajedrez. El término a menudo se usa indistintamente con sus subcampos, que incluyen el aprendizaje autónomo y el aprendizaje profundo. Sin embargo, hay ciertas diferencias. Por ejemplo, el aprendizaje automático se centra en la creación de sistemas que aprenden o mejoran su rendimiento en función de los datos que consumen. Es importante tener en cuenta que aunque todo aprendizaje automático es IA, no toda IA es aprendizaje automático.

Para obtener el valor completo de la IA, muchas empresas están haciendo inversiones significativas en equipos de ciencia de datos. La ciencia de datos, un campo interdisciplinario que usa métodos científicos y de otro tipo para extraer valor de los datos, combina conocimientos de campos como la estadística y la informática con el conocimiento empresarial para analizar los datos que se recopilan de múltiples fuentes.



Cómo la IA puede ayudar a las organizaciones

El principio fundamental de la IA es replicar, y luego superar, la forma en que los seres humanos perciben y reaccionan ante el mundo. Se está convirtiendo rápidamente en el cimiento principal de la innovación. La IA, impulsada por varias formas de aprendizaje automático que reconocen patrones en los datos para permitir predicciones, puede agregar valor a su negocio ya que permite:

- Proporcionar una comprensión más completa de la abundancia de datos disponibles
- Usar predicciones para automatizar las tareas excesivamente complejas o prosaicas

A en la empresa

La tecnología de IA está mejorando el rendimiento y la productividad de la empresa mediante la automatización de los procesos o las tareas que antes requerían del poder humano. La IA también puede dar sentido a los datos a una escala que ningún humano podría jamás. Esta capacidad puede generar importantes beneficios comerciales. Por ejemplo, [Netflix utiliza el aprendizaje autónomo para proporcionar un nivel de personalización](#) que ayudó a la compañía a aumentar su base de clientes en más del 25% en 2017.

Cómo las empresas usan la IA

De acuerdo con la *Harvard Business Review*, las empresas están utilizando la IA principalmente para:

Detectar y disuadir intrusiones de seguridad (44%)

Resolver problemas tecnológicos de los usuarios (41%)

Reducir el trabajo de la administración de producción (34%)

Medir el cumplimiento interno en el uso de proveedores aprobados (34%)



REFERENCIAS

Referencias en la Web:

[1]

<https://conceptodefinicion.de/programacion-informatica/>

[2]

[https://wiki.uqbar.org/wiki/articles/paradigma-de-programacion.html#:~:text=Un%20paradigma%20de%20programaci%C3%B3n%20es,relaciones%2C%20funciones%2C%20instrucciones\).](https://wiki.uqbar.org/wiki/articles/paradigma-de-programacion.html#:~:text=Un%20paradigma%20de%20programaci%C3%B3n%20es,relaciones%2C%20funciones%2C%20instrucciones).)

[3] <https://cibertareas.info/objetivos-de-las-redes-de-computadoras-redes-de-computadoras.html>

<http://reader.digitalbooks.pro/content/preview/books/37997/book/OEBPS/Text/chapter1.html>

http://www.redtauros.com/Clases/Fundamentos_Nets/01_Introduccion.pdf

https://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software#Objetivos

<https://kwfoundation.org/blog/2020/11/12/terminos-de-ia-inteligencia-artificial-que-necesitas-saber/>

<https://computerhoy.com/reportajes/tecnologia/inteligencia-artificial-469917>