

# Sistemas y Computación

## Systems and Computing

Autor: Johan Alejandro Martínez Martínez

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

Correo-e: j.martinez4@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.

### I.1 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 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.

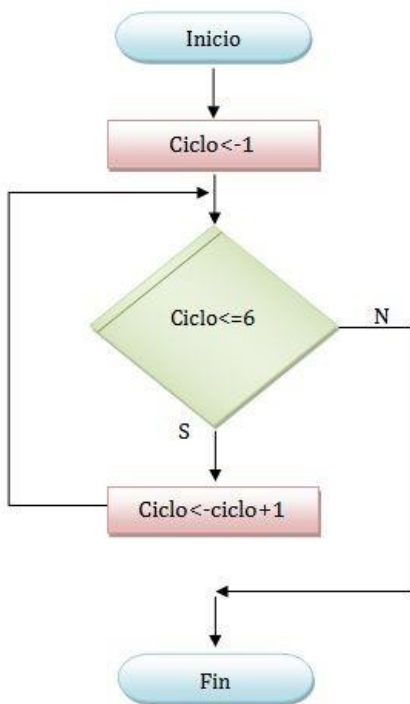


Figura 1. Paradigma estructurado

### PARADIGMA DE OBJETOS

El paradigma de objetos es una concepción en la cual se 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.

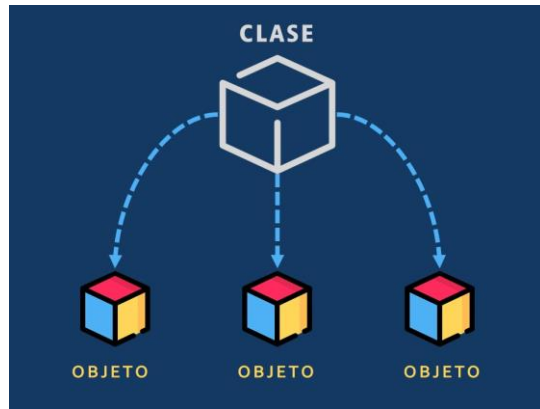


Figura 2. Paradigma orientado a objetos

### 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.

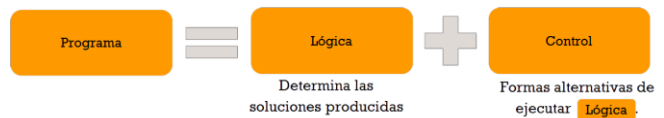


Figura 3. Paradigma lógico

### 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.

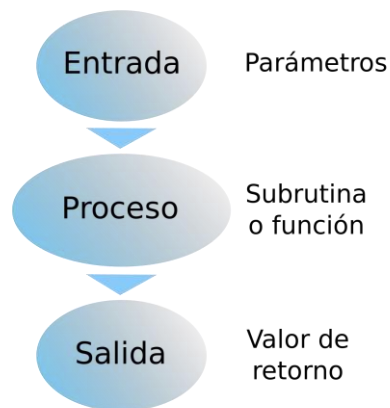


Figura 4. Paradigma funcional.

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.

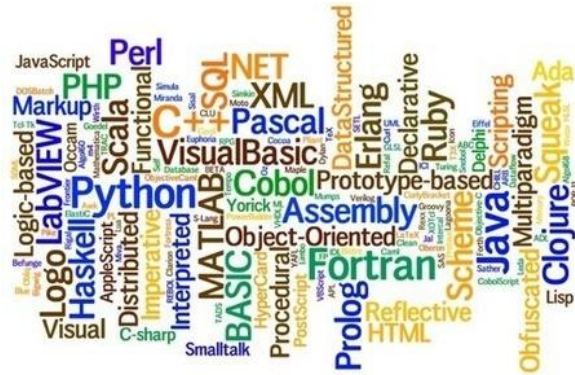


Figura 7. Lenguajes de programación.

## I.2 REDES Y COMUNICACIONES

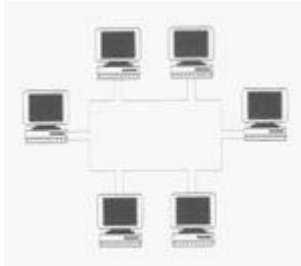
A mediados de los 70 diversos fabricantes desarrollaron sus propios sistemas de redes locales. Es en 1980 cuando Xerox, en cooperación con Digital Equipment Corporation e Intel, desarrolla y publica las especificaciones del primer sistema comercial de red denominado EtherNet. En 1986 IBM introdujo la red TokenRing. La mayor parte del mercado utiliza hoy día la tecnología del tipo EtherNet.

En 1982 aparecen los ordenadores personales, siendo hoy una herramienta común de trabajo. Esta difusión del ordenador ha impuesto la necesidad de compartir información, programas, recursos, acceder a otros sistemas informáticos dentro de la empresa y conectarse con bases de datos situadas físicamente en otros ordenadores, etc. En la actualidad, una adecuada interconexión entre los usuarios y procesos de una empresa u organización, puede constituir una clara ventaja competitiva. La reducción de costes de periféricos, o la facilidad para compartir y transmitir información son los puntos claves en que se apoya la creciente utilización de redes.

Una red es un conjunto de ordenadores conectados entre sí, que pueden comunicarse compartiendo datos y recursos sin importar la localización física de los distintos dispositivos. A través de una red se pueden ejecutar procesos en otro ordenador o acceder a sus ficheros, enviar mensajes, compartir programas...

Los ordenadores suelen estar conectados entre sí por cables. Pero si la red abarca una región extensa, las conexiones pueden realizarse a través de líneas telefónicas, microondas, líneas de fibra óptica e incluso satélites.

Cada dispositivo activo conectado a la red se denomina *nodo*. Un dispositivo activo es aquel que interviene en la comunicación de forma autónoma, sin estar controlado por otro dispositivo. Por ejemplo, determinadas impresoras son autónomas y pueden dar servicio en una red sin conectarse a un ordenador que las maneje; estas impresoras son nodos de la red.



### I.3 INGENIERÍA DE SOFTWARE

La ingeniería de software es una especialidad que consiste en sistemas, instrumentos y técnicas que se emplean en el desarrollo de los programas informáticos.

La ingeniería de software, también, incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema.

Se debe señalar, que el desarrollo del software va unido a lo que se conoce en el campo del software “ciclo de vida del software” que consiste en cuatro etapas que se conocen como: **concepción, elaboración, construcción y transición.**



**La concepción** determina la repercusión del proyecto y diseña el modelo de negocio; **la elaboración** precisa la planificación del proyecto, especificando las características y apoya la arquitectura; **la construcción** es la elaboración del producto; y **la transición** es la entrega del producto terminado a los usuarios.

El término **Ingeniería del Software**, surge cuando se llevaba a cabo dos conferencias estructuradas por la OTAN en los años en 1967 y 1968.

Es para mediados de la década los 70, cuando los sistemas informáticos incrementaron la dificultad y dan pie a la creación de las redes de computadoras.

Este hecho presionó a quienes desarrollaban las computadoras personalizadas, aunque no se sabía mucho sobre ellas; al final de esta etapa, surgieron los microprocesadores.

La cuarta época del progreso de los sistemas informáticos, comienza en los 90 y se orienta hacia el impacto general de las computadoras y el software, en todos los entornos.

La industria del software es de gran importancia en la economía mundial. Se ponen de manifiesto las técnicas de redes neuronales, al igual que la lógica difusa, de interés en el campo de la Inteligencia Artificial.

En la actualidad el software tiene un doble papel. Es el producto, pero al mismo tiempo, actúa como el conductor que entrega el producto. Como conductor utilizado para entregar el producto, actúa como base de control, por ejemplo un sistema operativo, o un sistema gestor de redes.

El software actúa como distribuidor y hace llegar a los usuarios, el producto más importante del este siglo: la información.

El software modifica la información personal para hacerlos más útiles en un entorno local, administra información comercial para mejorar la competitividad, facilita el acceso a redes a nivel mundial, y propone la forma de obtener información de cualquier manera.

En la actualidad la ingeniería del Software es considerada una nueva especialidad de la ingeniería y junto con la especialidad de Ingeniería Informática, es una de las profesiones con más demanda, aunque hay lugares en el mundo, en los que no es muy bien remunerada, como sucede en España.

La palabra ingeniería está estrechamente ligada a la de prestigio de lo que se deriva, que muchas ramas del conocimiento tiendan a autodenominarse así.

## I.4 INTELIGENCIA ARTIFICIAL

Hace tiempo abandonó el espectro de la ciencia ficción para colarse en nuestras vidas y, aunque todavía en una fase muy inicial, está llamada a protagonizar una revolución equiparable a la que generó Internet.



Las máquinas inteligentes imitan las funciones cognitivas de los humanos.

La Inteligencia Artificial (IA) es la **combinación de algoritmos planteados con el propósito de crear máquinas que presenten las mismas capacidades que el ser humano**. Una tecnología que todavía nos resulta lejana y misteriosa, pero que desde hace unos años está **presente en nuestro día a día a todas horas**.

### TIPOS DE INTELIGENCIA ARTIFICIAL

Los expertos en ciencias de la computación Stuart Russell y Peter Norvig diferencian varios tipos de inteligencia artificial:

- Sistemas que piensan como humanos: **automatizan actividades como la toma de decisiones, la**

**resolución de problemas y el aprendizaje**. Un ejemplo son las redes neuronales artificiales.

- Sistemas que actúan como humanos: se trata de computadoras que **realizan tareas de forma similar a como lo hacen las personas**. Es el caso de los robots.
- Sistemas que piensan racionalmente: intentan **emular el pensamiento lógico racional de los humanos**, es decir, se investiga cómo lograr que las máquinas puedan percibir, razonar y actuar en consecuencia. Los sistemas expertos se engloban en este grupo.
- Sistemas que actúan racionalmente: idealmente, son aquellos que tratan de **imitar de manera racional el comportamiento humano**, como los agentes inteligentes.

### REFERENCIAS

#### Referencias en la Web:

[1] <https://conceptodefinicion.de/programacioninformatica/>

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

<https://www.monografias.com/trabajos58/redes-comunicaciones/redes-comunicaciones.shtml>

<https://micarrerauniversitaria.com/c-ingenieria/ingenieria-de-software/>

<https://www.iberdrola.com/innovacion/que-es-inteligencia-artificial>

