

Sistemas y Computación

Systems and Computing

Autor: Juan Camilo Rodríguez Libreros
IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia
Correo-e: juancamilo.rodriguez1@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.

1.1 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 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 descriptas 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.

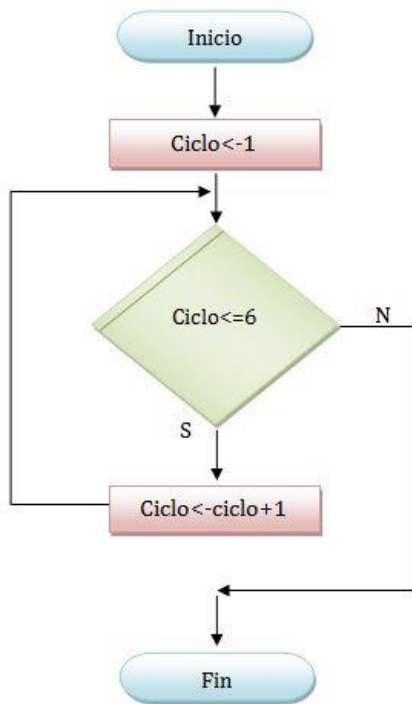


Figura 1. Paradigma estructurado

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.

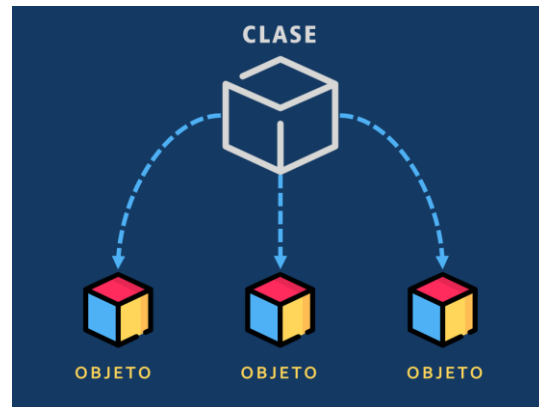


Figura 2. Paradigma orientado a objetos

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.

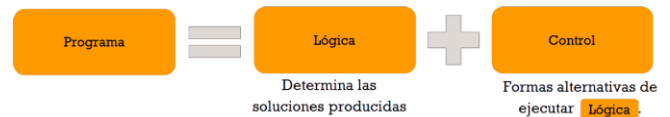


Figura 3. Paradigma lógico

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.

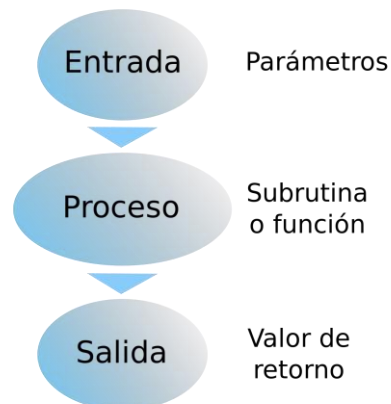


Figura 4. Paradigma funcional.

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.

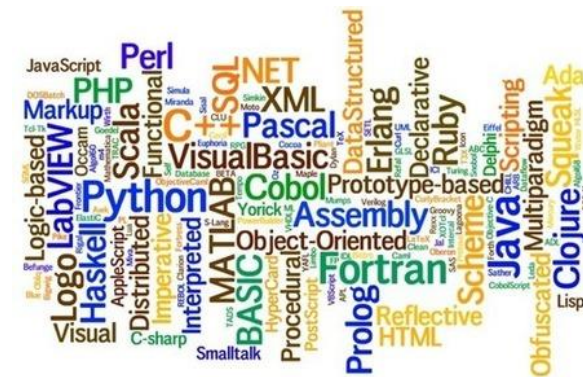


Figura 7. Lenguajes de programación.

1.2 REDES Y COMUNICACIONES

Este se puede definir como un conjunto de ordenadores conectados entre sí en donde se pueden enviar archivos, ficheros, programas etc. Estas conexiones normalmente se hacen por medio de cables; si la red es muy extensa se utilizan líneas telefónicas, cables de fibra óptica, microondas o satélites. Cada dispositivo conectado a la red se le denomina “nodo” o “servidor”.

Las redes se pueden clasificar en:

LAN: local area network, estas abarcan zonas pequeñas por ejemplo edificios o campus

WAN: wide area network, esta abarca zonas mas grandes puede llegar a abarcar muchos países.



Figura 8. Red

MEDIO DE TRANSMISIÓN

El medio de transmisión es la parte física en la cual se reparten datos de nodo a nodo. Como se decía anteriormente los medios más usados son los cables, también se utiliza el medio inalámbrico pero la desventaja es que es un poco más lento; los tipos de cables utilizados son:

CABLE COAXIAL

Hay distintos cables coaxiales que se usan dependiendo de la distancia, protección etc. Estos son:

Cable thick o grueso: es un poco caro y difícil de instalar, pero es bueno para conectar varios servidores y cubrir largas distancias.

Cable thin o fino: este es más barato y fácil de instalar, pero conecta un número reducido de servidores.



Figura 9. Cable coaxial

CABLE PAR TRENZADO

Este consta de dos conductores aislados trenzados entre ellos protegidos con un aislante, puede haber entre 2-3 pares en el cable. Este cable se utiliza en la telefonía, es muy barato y fácil de instalar, la única desventaja que tiene es que es muy susceptible a interferencias eléctricas.

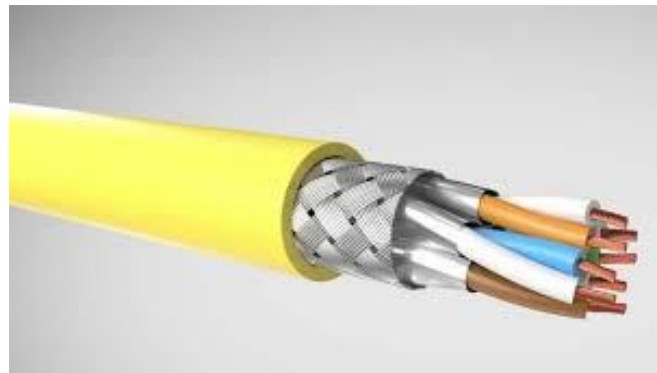


Figura 10. Cable par trenzado

FIBRA ÓPTICA

Este es el medio mas avanzado y moderno, permite tener conexiones muy rápidas y transmitirla a través de kilómetros de distancia sin necesidad de amplificar la señal, no es susceptible a interferencias eléctricas; por estas virtudes es usado para redes que requieren velocidad y son extensas; el inconveniente es que el equipo para instalarlo es muy caro y se necesitan técnicos calificados para hacer la instalación y el mantenimiento.

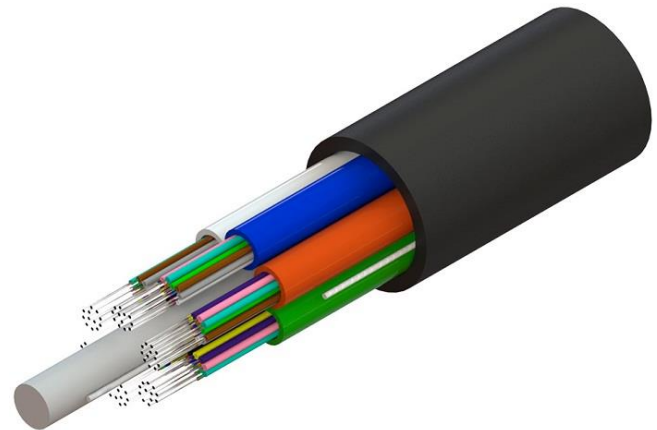


Figura 11. Fibra óptica

ANCHO DE BANDA

El método de transmisión hace relación a la capacidad del medio para transmitir información. El ancho de banda es la diferencia de la frecuencia que tiene una onda, entre mas alta es la frecuencia más rápida es la comunicación.

TOPOLOGIA DE RED

Se refiere a como es la forma de la red, las topologías mas usadas son, la de bus, anillo, estrella y árbol

BUS

Es un diseño en el que un cable “bus” es compartido entre todos los nodos. La ventaja que tiene esta red es su sencillez y economía, la desventaja es que si el cable falla toda la red se colapsa.



Figura 12. Red en bus

ANILLO

Este tiene un diseño circular, es unidireccional esto quiere decir que los datos circulan en un solo sentido. La ventaja que tiene es que es fácil de instalar y reconfigurar, su desventaja es que como es unidireccional la información tiene que pasar nodo por nodo.

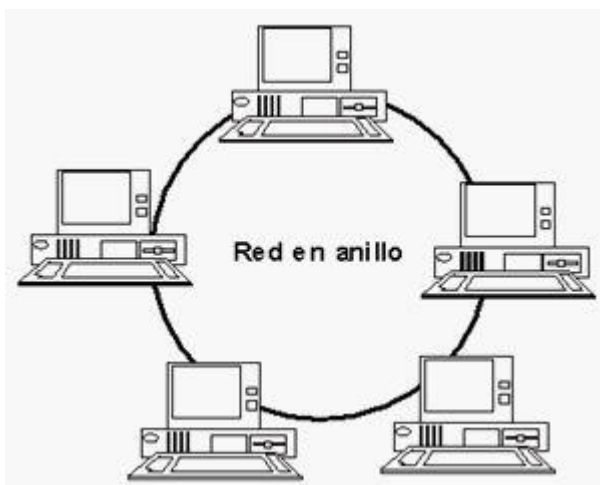


Figura 13. Anillo

ESTRELLA

El diseño es como el de una estrella, cada nodo está conectado a un HUB o caja de conexiones por lo que tiene un cable independiente. La ventaja es que si un cable falla solo afectará a un nodo, la desventaja es que si el HUB falla toda la red cae, también es un poco costosa por la cantidad de cableado.

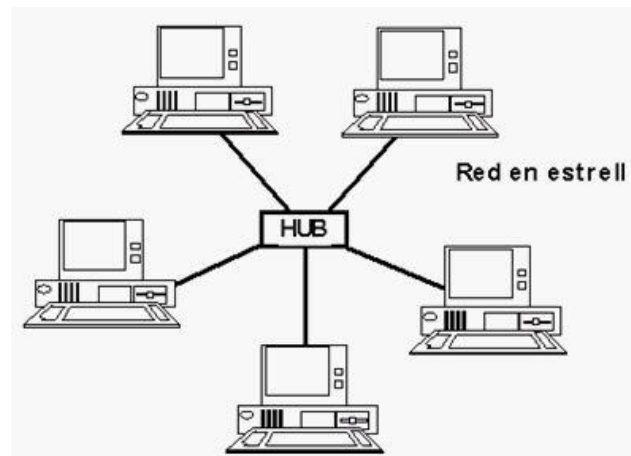


Figura 14. Estrella

ARBOL

Esta red también se le llama estrella distribuida, al igual que la topología en estrella cada nodo se conecta a un HUB que está conectado a un cable en “bus” y este reparte la información.

Reúne las ventajas y desventajas de las topologías en bus y estrella.

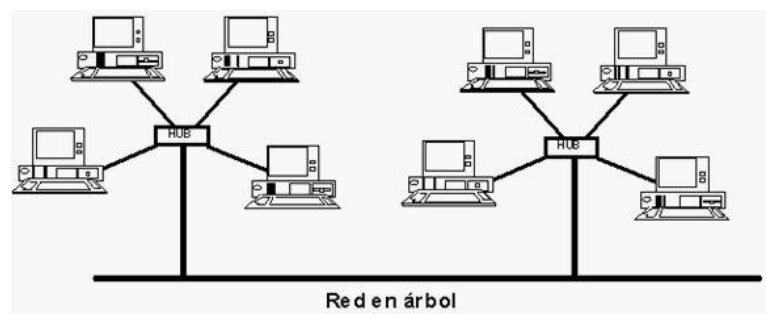


Figura 15. Árbol

1.3 INGENIERÍA DE SOFTWARE

Esta es una subrama de la ciencia computacional que busca crear software seguro y de calidad basándose en técnicas de ingeniería, el término ingeniería de software se adoptó en el final de la década del 50.

OBJETIVOS

Como se describía anteriormente se busca crear software seguro y de calidad, para poder implementarlo de acuerdo a las necesidades de una compañía y facilitar por ende su trabajo, también el de mejorar otros softwares previamente creados.

RECURSOS

Para poder crear un buen software se necesita de personas experimentadas (ingenieros informáticos y programadores), como también un hardware de alta calidad para poder desarrollar o crear este.

LA NATURALEZA DE LA INGENIERIA DE SOFTWARE

Como es una disciplina que esta orientada a usar técnicas de ingeniería para crear software de calidad se necesitan las matemáticas para crear algoritmos, necesita también una creación progresiva para que cada parte del proyecto funcione a la perfección y finalmente una buena gestión del proyecto pues requiere de presupuestos, ciertos tiempos de entrega y de un equipo de profesionales para dirigir.

¿PORQUE ES IMPORTANTE?

En el mundo actual estamos acostumbrados a utilizar la tecnología en diferentes ámbitos de la vida, utilizamos computadores, redes sociales, celulares, maquinas etc. un buen software va a hacer que todos estos funcionen conforme a nuestras necesidades y además nos va a garantizar calidad y seguridad del funcionamiento de estas.



Figura 16. Gráfico ingeniería de software

1.4 INTELIGENCIA ARTIFICIAL

Se define como la inteligencia llevada a cabo por maquinas o computadores.

TIPOS DE INTELIGENCIA ARTIFICIAL

Hay diferentes tipos de esta y son:

Sistemas que piensan como humanos: son sistemas que tratan de emular el pensamiento de una persona por ejemplo las redes neuronales artificiales.

Sistemas que actúan como humanos: son sistemas que simulan el comportamiento humano por ejemplo los robots.

Sistemas que piensan racionalmente: estos sistemas imitan el pensamiento lógico de un humano por ejemplo los sistemas expertos.

Sistemas que actúan racionalmente: son sistemas que imitan de forma racional el comportamiento humano por ejemplo los agentes inteligentes.

APLICACIONES DE LA INTELIGENCIA ARTIFICIAL

Tiene muchos campos de aplicación algunos de ellos son: la robótica, mecatrónica, videojuegos, medicina, industria entre otros.

Debido a que cada campo requiere una solución a un problema la inteligencia artificial ha ayudado a resolver o facilitar las tareas de todos estos.

Nota: un ingeniero de software puede programar o crear un software inteligente.

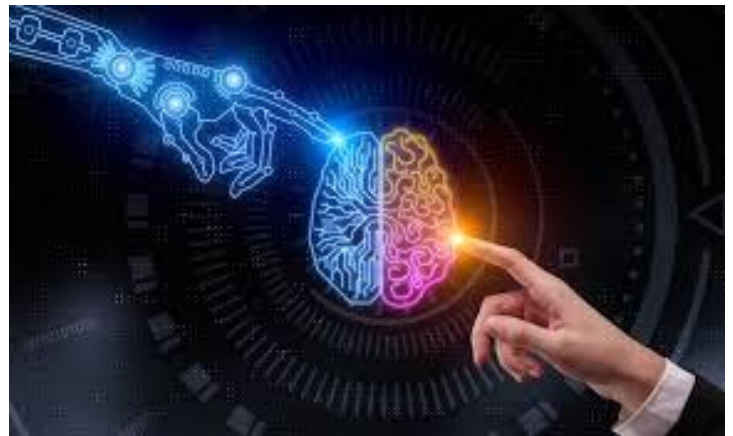


Figura 17. Ilustración inteligencia artificial y humana.

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

[4] https://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software

[5] https://es.wikipedia.org/wiki/Inteligencia_artificial