**COLEGIO VOCACIONAL DE ARTES Y OFICIOS DE CARTAGO NOCTURNO**

**CURSO LECTIVO 2021**

**Primer Periodo**

**Enunciado Actividad Virtual ·1**

**NIVEL: DECIMO**

**SECCIÓN: 10-05**

**Sub-Área: TI**

**PROFESOR**

**Lic. Alexander Monge Vargas**

**Alumno: Aaron Walker Uba**

Contenido

[Introducción 3](#_Toc69571770)

[Marco Teórico. 3](#_Toc69571771)

[Objetivos Generales: 4](#_Toc69571772)

[Objetivos Específicos: 4](#_Toc69571773)

[Desarrollo 5](#_Toc69571774)

[CAPITULO 1 5](#_Toc69571775)

[Algoritmo 5](#_Toc69571776)

[Características de un Algoritmo 5](#_Toc69571777)

[Tipos de algoritmos. (En que los usamos) 5](#_Toc69571778)

[CAPITULO 2 7](#_Toc69571779)

[Pseudocodigo 7](#_Toc69571780)

[Softwares para hacer un pseudocodigo: 7](#_Toc69571781)

[CAPITULO 3 8](#_Toc69571782)

[Diagramas de flujo 8](#_Toc69571783)

[software para crear diagramas de flujo y mapas mentales: 8](#_Toc69571784)

[9](#_Toc69571785)

[CAPITULO 4 10](#_Toc69571786)

[Leguaje de programación 10](#_Toc69571787)

[Clasificación de lenguajes de programación. 10](#_Toc69571788)

[Ejemplos de lenguajes de programación. 11](#_Toc69571789)

[1. Conclusiones 12](#_Toc69571790)

[Referencias Bibliográficas 13](#_Toc69571791)

[Anexos i](#_Toc69571792)

Introducción

Marco Teórico.

En 1944 se crea la Mark I, una idea basada en el invento del francés Charles Jacquard un fabricante de tejidos quien logró producir automáticamente patrones de tejidos leyendo información codificado en patrones de agujeros perforados en tarjetas de papel rígido.

Los primeros computadores **surgieron como máquinas de cálculo lógico**, debido a las necesidades de los aliados durante la [Segunda Guerra Mundial](https://www.caracteristicas.co/segunda-guerra-mundial/).

**En febrero de 1951 apareció la Ferranti Mark 1**, una versión moderna de la computadora norteamericana del mismo nombre que estaba disponible comercialmente. Fue sumamente importante en la historia del computador, pues contaba con un índice de registros, que permitía la [lectura](https://www.caracteristicas.co/lectura/) más fácil de un conjunto de palabras en la memoria. En 1953 se establece como el primer lenguaje formal de programación, o sea, el primer programa diseñado para fabricar programas computacionales, por los programadores de IBM, liderados por John Backus.

**Inicialmente se desarrolló para el computador IBM 704**, y para una variada gama de aplicaciones científicas y de ingeniería, razón por la cual tuvo una amplia serie de versiones a lo largo de medio siglo de implementación.

La primera computadora moderna apareció en otoño de 1968, como un prototipo presentado por Douglas Engelbart. **Tenía por primera vez un ratón o puntero, y una interfaz gráfica de usuario**

La primera [red de computadores](https://www.caracteristicas.co/redes-de-computadoras/) del mundo**fue ARPANET, creada en 1968** por el Departamento de Defensa de los Estados Unidos. Sirvió de plataforma rápida de intercambio de información entre instituciones educativas y estatales, con fines probablemente militares.

Esta red se desarrolló, actualizó y eventualmente **se convirtió en la columna vertebral de**[Internet](https://www.caracteristicas.co/internet/), abierta ya al público en general, al menos hasta 1990.

Las computadoras h**oy son parte de la vida cotidiana**, a punto tal que para muchos es inconcebible ya un mundo sin ellas. Se las encuentra en nuestras oficinas, en nuestros [teléfonos celulares](https://www.caracteristicas.co/telefono/), en diversos electrodomésticos, a cargo de instalaciones automatizadas, y desempeñando un sinfín de operaciones de manera automática e independiente.

Esto**tiene muchos aspectos positivos, pero también entraña muchos temores**. Por ejemplo, el surgimiento de la robótica, paso siguiente natural de la computación, promete dejar sin empleo a muchos trabajadores humanos, superados por la capacidad de automatización que cada día es mayor y más rápida

Objetivos Generales:

Conocer todos los conceptos básicos para poder iniciar en el mundo de la computación y de esa forma entender y para futuras lecciones comprender de manera más fácil los conceptos y el funcionamiento

Objetivos Específicos:

* Conocer lo que es un algoritmo
* Conocer lo que es un pseudocodigo
* Conocer el concepto de diagrama de flujo
* Conocer el concepto de lenguaje de programación

Desarrollo

# CAPITULO 1

# Algoritmo

Un algoritmo es un procedimiento paso a paso para conseguir un fin. Es un concepto comúnmente relacionado con el ámbito de la informática. Un algoritmo informático es un conjunto de instrucciones definidas, ordenadas y acotadas para resolver un problema o realizar una tarea.

En programación, supone el paso previo a ponerse a [escribir el código](https://www.instagram.com/p/CGkHG_lg8xY/).

## Características de un Algoritmo

los algoritmos presentan una serie decaracterísticas comunes. Son:

* **Precisos**. Objetivos, sin ambigüedad.
* **Ordenados**. Presentan una secuencia clara y precisa para poder llegar a la solución.
* **Finitos**. Contienen un número determinado de pasos.
* **Concretos**. Ofrecen una solución determinada para la situación o problema planteados.
* **Definidos**. El mismo algoritmo debe dar el mismo resultado al recibir la misma entrada.

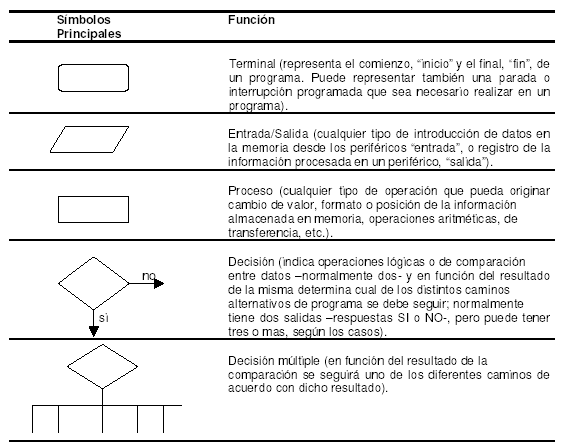
## Tipos de algoritmos. (En que los usamos)

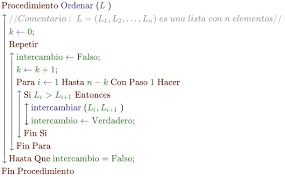
**Algoritmos de búsqueda**: Los **algoritmos de búsqueda** localizan uno o varios elementos que presenten una serie de propiedades dentro de una estructura de datos.

**Algoritmos de ordenamiento:** Reorganizan los elementos de un listado según una relación de orden.

**Algoritmos voraces:** Los **algoritmos voraces** consisten en una estrategia de búsqueda que sigue una heurística en la que se elige la mejor opciónóptima en cada paso local con el objetivo de llegar a una solución general óptima.

**Algoritmos probabilísticos:** Es una técnica que usa una fuente de aleatoriedad como parte de su lógica. Mediante un muestreo aleatorio de la entrada llega a una solución que puede no ser totalmente óptima, pero que es adecuada para el problema planteado.

Símbolos y funciones de los diagramas



Ejemplo de pseudocodigo

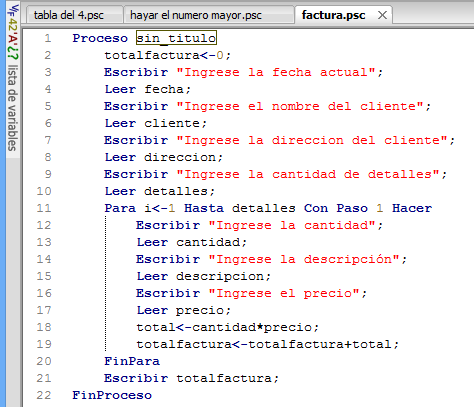
# CAPITULO 2

# Pseudocodigo

El **pseudocódigo**es una forma de expresar los distintos pasos que va a realizar un programa, de la forma más parecida a un lenguaje de programación. Su principal función es la de representar por pasos la solución a un problema o algoritmo, de la forma más detallada posible, utilizando un lenguaje cercano al de programación.

## Softwares para hacer un pseudocodigo:

* Pseint: es una herramienta para asistir a un estudiante en sus primeros pasos en programación mediante un simple e intuitivo pseudolenguaje en español (complementado con un editor de [diagramas de Flujo](https://www.areatecnologia.com/diagramas-de-flujo.htm)).

Ejemplo 2 de pseudocodigo

Ejemplo 3 de pseudocodigo

# 

# CAPITULO 3

# Diagramas de flujo

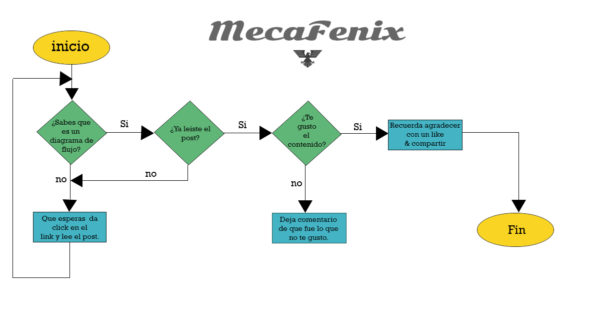
Un diagrama de flujo es la representación gráfica del flujo o secuencia de rutinas simples. Tiene la ventaja de indicar la secuencia del proceso en cuestión, las unidades involucradas y los responsables de su ejecución, es decir, viene a ser la representación simbólica o pictórica de un procedimiento administrativo.

Hay tres clases de diagramas que generalmente se utilizan según las necesidades o bien el propósito para el cual se le requiera:  å Diagrama de Bloques ç Diagrama Gráfico é Diagrama de Flujo de Proceso

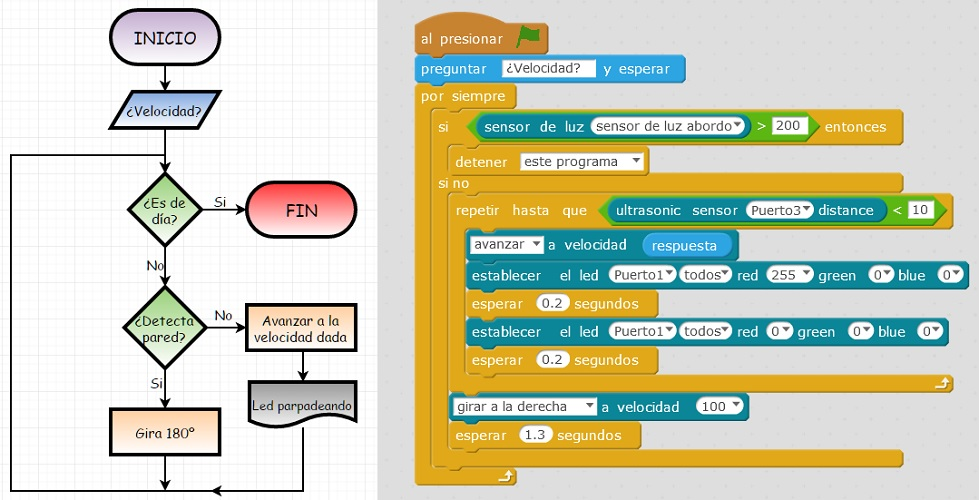
## software para crear diagramas de flujo y mapas mentales:

* LucidChart. Característica favorita. Planes y precios.
* Microsoft Visio. Característica favorita. Planes y precios.
* Creately. Característica favorita. Planes y precios.
* Pidoco.
* Google Docs.
* Raptor

Ejemplo 1 de diagrama de flujo



Ejemplo 2 de diagrama de flujo



# CAPITULO 4

# Leguaje de programación

**Es un programa destinado a la construcción de otros**[programas informáticos](https://concepto.de/programa-informatico/). Su nombre se debe a que comprende un lenguaje formal que está diseñado para organizar [algoritmos](https://concepto.de/algoritmo-en-informatica/) y procesos lógicos que serán luego llevados a cabo por un ordenador o sistema informático, permitiendo controlar así su comportamiento físico, lógico y su [comunicación](https://concepto.de/comunicacion/) con el usuario humano.

La máquina del telar creada por Jacquard, en el año de 1801, empleó los huecos en tarjetas que tenían perforaciones para recrear el movimiento del brazo mecánico de una máquina de tejer, con el objeto de crear patrones de decoración de forma automática. Herman Hollerith codificó toda la información de las tarjetas perforadas, cuando se percató que los choferes de los trenes lograban identificar los pasajeros usando el hueco que hacían en el boleto. En los años 40 del siglo XX se desarrollaron los primeros ordenadores, con alimentación de corriente eléctrica. Tenían limitaciones de velocidad y la capacidad de memoria, obligando a los programadores a crear programas sencillos. Los años de 1980 C++, llegó para conjugar la programación hacia los objetos y la programación de sistemas. C++, llegó para conjugar la programación hacia los objetos y la programación de sistemas.

## Clasificación de lenguajes de programación.

* **Lenguajes de bajo nivel.** Se trata de lenguajes de programación que están diseñados para un hardware específico y que por lo tanto no pueden migrar o exportarse a otros [computadores](https://concepto.de/computador/). Sacan el mayor provecho posible al sistema para el que fueron diseñados, pero no aplican para ningún otro.
* **Lenguajes de alto nivel.** Se trata de lenguajes de programación que aspiran a ser un lenguaje más universal, por lo que pueden emplearse indistintamente de la arquitectura del [hardware](https://concepto.de/hardware/), es decir, en diversos tipos de sistemas. Los hay de propósito general y de propósito específico.
* **Lenguajes de nivel medio.** Este término no siempre es aceptado, que propone lenguajes de programación que se ubican en un punto medio entre los dos anteriores: pues permite operaciones de alto nivel y a la vez la gestión local de la arquitectura del sistema.

## Ejemplos de lenguajes de programación.

* **BASIC.** Su nombre proviene de las siglas de *Beginner’s All- purpose Symbolic Instruction Code* (Código simbólico de instrucciones de propósito general para principiantes), y es una familia de lenguajes imperativos de alto nivel, aparecidos por primera vez en 1964. Su versión más actual es Visual Basic .NET.
* **COBOL.** Su nombre es un acrónimo para *Common Business-Oriented Lenguage* (Lenguaje común orientado a los negocios) y se trata de un lenguaje de programación universal creado en 1959, orientado principalmente a la informática de gestión, es decir, empresarial.
* **FORTRAN.** Su nombre proviene de *The IBM Mathematical Formula Translating System*(El sistema de traducción de fórmulas matemáticas de IBM), y es un lenguaje de programación de alto nivel, propósito general y de tipo imperativo, diseñado para aplicaciones científicas y de ingeniería.
* **Java.**Un lenguaje de programación de propósito general, orientado a objetos, cuyo espíritu se resume en las siglas WORA: *Written Once, Run Anywhere*, es decir: Escrito una vez, funciona en cualquier parte. La idea era diseñar un lenguaje universal empleando sintaxis derivada de los lenguajes C y C++, pero empleando menos utilidades de bajo nivel que cualquiera de ambos.

## 

1. Conclusiones

Este informe sirvió para poder iniciar en el camino de la informática, conceptos básicos que ayudarán a todos al momento de iniciar a estudiar cualquier carrera relacionada a la informática, hoy en día la informática es muy importante en cualquier carrera que exista y estudiar algo dedicado a la informática trae muchos beneficios a nivel laboral, la sociedad se actualiza poco a poco, y nosotros debemos actualizarnos también, a comprender todo los conceptos necesarios y entender bien lo que hacemos a la hora de crear algún pseudocodigo, diagrama de flujo o cualquier programa informático o pagina web que debamos hacer a lo largo de nuestra carrera, la informática es importante en este mundo así que hay que aprender todo lo necesario y saber lo que estamos haciendo.

Los diagramas de flujo y pseudocódigos tienen gran importancia a la hora de realizar algún programa o página, ya que estructura y nos guía al crear el código, no solo cuando estamos aprendiendo, también a la hora de ser profesionales, estos diagramas de flujo y pseudocodigo son importantes en cualquier momento, y es bueno saber lo que conlleva.

Referencias Bibliográficas

"Historia de la Computadora". Autor: Julia Máxima Uriarte. Para: Caracteristicas.co. Última edición: 24 de noviembre de 2020. Disponible en: <https://www.caracteristicas.co/historia-de-la-computadora/>.

“Que es un algoritmo en la informática”. Autor: Raquel Manuela de Vega. Para: Profile.es. Última edición: 01 de abril del 2021. Disponible en: https://profile.es/blog/que-es-un-algoritmo-informatico/

“Que es pseudocodigo”. Autor: Ángel Robledano. Para: openwebinars.net. Ultima edición: 18 de junio de 2019.Disponible en: <https://openwebinars.net/blog/que-es-pseudocodigo/>

Manene, L. M. (2011). Los diagramas de flujo: su definición, objetivo, ventajas, elaboración, fases, reglas y ejemplos de aplicaciones. *Recuperado el*, *5*.

Última edición: 11 de julio de 2020. Cómo citar: "Lenguaje de Programación". Autor: María Estela Raffino. De: Argentina. Para: *Concepto. De*. Disponible en: https://concepto.de/lenguaje-de-programacion/. Consultado: 17 de abril de 2021.

[Yohana Agüero](https://tecnoinformatic.com/author/yohana/)(mayo, 2020). “Historia de los lenguajes de programación” Para: tecnoinformatic.com. Disponible en: https://tecnoinformatic.com/c-programacion/historia-de-los-lenguajes-de-programacion/

Anexos

