



UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO



FACULTAD DE INGENIERÍAS

FUNDAMENTOS DE PROGRAMACIÓN

Tarea 1: Historia de la programación

Nombre: González Ramírez Octavio Alberto

Fecha: 05/Octubre/2020

Historia del cómputo

Uno de los primeros instrumentos para realizar cálculos fue el ábaco, consta de dos palos de madera (actualmente puede ser de diferente material) que sostienen algunas varillas, en las cuales están incrustadas diez cuentas en cada una. Éste se remonta en las antiguas civilizaciones griega y romana.

Otro invento importante fue creado por el francés Blaise Pascal, el cual era la *Pascalina* creada en 1642. Con esta máquina los datos eran representados de acuerdo con la posición de los engranajes y las posiciones finales de las ruedas eran establecidas gracias a que los datos se introducían de manera manual. El cuentakilómetros que actualmente tienen instalados los automóviles están basados con el funcionamiento de la Pascalina.

En el siglo XIX se creó la primera máquina analítica, considerada la primera computadora, fabricada por *Charles Babbage*, ingeniero inglés y profesor matemático de la Universidad de Cambridge. En 1823 el gobierno británico lo apoyó para crear el proyecto de una máquina de diferencias, un dispositivo mecánico para efectuar sumas repetidas. La idea que tuvo Charles Babbage sobre un computador nació debido a que la elaboración de las tablas matemáticas era un proceso tedioso y propenso a errores. Las características de esta máquina incluyen una memoria que puede almacenar hasta 1000 números de hasta 50 dígitos cada uno. Las operaciones por ejecutar por la unidad aritmética son almacenadas en una *tarjeta perforadora*. Se estima que la máquina tardaría un segundo en realizar una suma y un minuto en una multiplicación.

En la década de 1880 Estados Unidos deseaba agilizar el proceso del censo de 1890. Para lograrlo la oficina de censos contrató al estadístico *Herman Hollerith* para que diseñara una técnica que pudiera lograrlo. Hollerith planificó un sistema basado en que los datos serían perforados en tarjetas de acuerdo con un formato preestablecido. Ya que fueran perforadas las tarjetas éstas serían tabuladas y clasificadas por una máquina basada en el telar de *Joseph Jacquard*. Esta máquina realizaba el diseño gracias a la información obtenida con las tarjetas.

Para el año de 1944 en la Universidad de Harvard se construyó la *Mark I*, diseñada por un equipo encabezado por *Howard H. Aiken*. Ésta máquina demoraba de seis segundos para realizar una multiplicación y el doble para una división. Las dimensiones de este computador eran de 17 metros de largo, 3 metro de alto y 1 metro de profundidad, estaba basado en rieles, aproximadamente 3000 rieles con alrededor de 800 kilómetro de cable. Al Mark I se le hicieron mejoras sucesivas, obteniendo así el *Mark II*, *Mark III* y *Mark IV*.

Hacia el año de 1947 en la Universidad de Pennsylvania se construyó la primera computadora electrónica, la *Electronic Numerical Integrator and Calculator* (ENIAC) que funcionaba con tubos al vacío y el diseño estuvo a cargo de los ingenieros *John Mauchly* y *John Eckert*. En este computador fueron incluidas nuevas técnicas de la electrónica que permitían disminuir la utilización de partes mecánicas. Debido a esto, la ENIAC superó ampliamente al *Mark I*, ya que podía efectuar alrededor de 500 sumas o 500 multiplicaciones en un segundo, además de que permitía el uso de aplicaciones científicas en astronomía, meteorología, etc.

Durante el desarrollo del proyecto ENIAC, el matemático Von Neumann propuso unas mejoras que ayudaron a llegar a los modelos actuales de computadoras:

1. Utilizar un sistema de numeración de base dos (Binario) en vez del sistema decimal tradicional.
2. Hacer que las instrucciones de operación estén en la memoria, al igual que los datos. De esta forma, memoria y programa residirán en un mismo sitio.

En 1949 se construyó la *Electronic Discrete Variable Automatic Computer* (EDVAC) en la Universidad de Manchester en Estados Unidos. fue el primer equipo con capacidad de almacenamiento de memoria e hizo desear a los otros equipos que tenían que ser intercambios o reconfigurados cada vez que se usaban. Tenía aproximadamente cuatro mil bulbos y usaba un tipo de memoria basado en tubos llenos de mercurio por donde circulaban señales eléctricas sujetas a retardos.

Historia de la programación

La historia de la programación está relacionada directamente con la aparición de las computadoras, cuyos datos se remontan en el siglo XV con la construcción de una máquina que realizaba operaciones básicas.

Gracias a que los avances en las ciencias de la informática fueron muy acelerados se reemplazaron los tubos al vacío por transistores en el año de 1958. El mismo año se sustituyeron por circuitos integrados, y en 1961 se miniaturizaron en chips de silicio. Hacia el año de 1971 *Intel* mostró su primer microprocesador y en 1973 surgió el primer sistema operativo CP/M.

La programación también tiene como influencia la máquina diferencial para cálculo de polinomios, proyecto que no fue concluido de *Charles Babbage* en apoyo de *Lady Ada Countess of Lovelace*, quien fuese la primera persona que se adentró en el lenguaje de la programación y de quien proviene el lenguaje de programación ADA creado por el departamento de defensa de Estados Unidos en la década de 1970.

Inicialmente la programación estaba basada en el código binario, el cual es una cadena de ceros y unos que además es el lenguaje que entiende de manera directa un computador. Después surgió el lenguaje ensamblador que, aunque fuese similar al código binario, al estar con letras era más fácil recordarlo. Posteriormente aparecieron lenguajes de nivel más alto que utilizan palabras en inglés generalmente, esto para dar las órdenes a seguir, para lo cual utilizan un proceso intermedio entre el lenguaje y el nuevo código llamado *Código Fuente*, el cual es un proceso que puede ser un compilador o intérprete.

Un compilador lee todas las instrucciones y genera un resultado; un intérprete ejecuta y genera resultados línea a línea. En cualquier caso, han aparecido nuevos lenguajes de programación, unos denominados estructurados y en la actualidad, en cambio, los lenguajes orientados a objetos y los lenguajes orientados a eventos.

Lista de lenguajes de programación

- **PHP** es un lenguaje de programación de código abierto que se enfoca para el desarrollo web. Entre las cosas que se pueden hacer la creación de dinámicas web al poderse incrustar dentro de HTML y su uso como *scripts* del lado del servidor.
- **Java** es un lenguaje que se destaca por ser multiplataforma. Su código tiene la ventaja de que se puede ejecutar en cualquier dispositivo. Es uno de los lenguajes más utilizados actualmente gracias a su función multiplataforma.
- **Python** es un lenguaje muy versátil de código abierto. Se utiliza principalmente para la inteligencia artificial y el Big data. Es un lenguaje sencillo, legible y elegante, dispone de un período de aprendizaje muy corta, además de que puede ser utilizada en cualquier sistema operativo actual.
- **C/C++** el lenguaje C, y su extensión C++ sigue siendo uno de los lenguajes más utilizados actualmente. Se utiliza cuando es necesaria una programación muy potente, en el cual a veces es necesario programar algunas tareas en el ensamblador. A pesar de tener una curva de aprendizaje muy alta, sigue siendo muy demandado en la programación en general, y particularmente en los videojuegos.
- **Javascript** es un lenguaje muy utilizado en la programación web del lado del cliente. Además de ser muy práctico es capaz de ser ejecutado en cualquier navegador, por lo que lo convierte en el lenguaje más utilizado en la web. Es un lenguaje multiplataforma, está orientado a objetos y eventos, además de ser interpretado.
- **C# y Visual Basic** son dos de los lenguajes orientados a objetos de Microsoft que utilizan su *framework.net*. son utilizados para crear una gran variedad de aplicaciones.
- **Objective-C** es un lenguaje basado en C y se destina al desarrollo de aplicaciones de iOS y macOS. Es un lenguaje orientado a objetos con el que se desarrollan la mayoría de apps para iPhone y iPad.

- **Otros lenguajes de programación** se destacan otros lenguajes que se utilizan actualmente:
 1. **Ruby** es interpretado donde todo es un objeto.
 2. **Swift** es utilizado para iOS y macOS
 3. **SQL** se utiliza para obtener información de una base de datos
 4. **Delphi** object pascal
 5. **Perl** se utiliza para aplicaciones web CGI

Clasificación de los lenguajes de programación

- **Editores de texto** sirven para crear y gestionar archivos digitales compuestos por texto. Cumplen con la función de leer el archivo e interpretar los bytes según el código del editor.
- **Compilador** es una especie de traductor que edifica el lenguaje de programación a un programa en sí.
- **Intérprete** analiza y transforma información sobre otros programas
- **Enlazador** toma los objetos iniciales del proceso de compilación, descarta los no necesarios y enlaza un código abierto con la biblioteca interna que produce un archivo ejecutable.
- **Depuradores** sirven para borrar o limpiar los errores de otros programas
- **IDE** los entornos de desarrollo integrado a una aplicación que le brinda al programador servicios integrales para facilitar el desarrollo de software.

Tipos de paradigmas de programación

un paradigma de programación es una propuesta tecnológica adoptada por un grupo de programadores cuyo núcleo central es incuestionable en cuanto a que unívocamente trata de resolver problemas delimitados. La solución de los problemas debe superponer consecuentemente un avance significativo en al menos un parámetro que afecte a la ingeniería en cómputo.

Los tipos de paradigmas son:

- **Programación imperativa**
- **Programación funcional**
- **Programación lógica**
- **Declarativo**
- **POO Orientado a objetos**
- **Por procedimientos**

Normalmente se mezclan todos los tipos de paradigmas a la hora de programar. De esta manera se da origen a la programación multiparadigma, pero los paradigmas que son más utilizados son los paradigmas orientados a objetos.

Bibliografía:

Cervantes N. & Pineda, C. (s. f.) *Un poco de historia*. Fundamentos de programación. Recuperado 4 de octubre de 2020, de http://www.utn.edu.ec/reduca/programacion/fundamentos/un_poco_de_historia.html

Computación Aplicada al Desarrollo S.A. de C.V. (s. f.). *Historia de la Computación*. Computación para estudiantes. Recuperado 4 de octubre de 2020, de http://www.cad.com.mx/historia_de_la_computacion.htm

Trejos O. (2018). *Aprovechamientos de los tipos de pensamiento matemático en el aprendizaje de la programación funcional*. Tecnura, 22(56), 29-39
<https://doi.org/10.14483/22487638.12807>