Logotipo, nombre de la empresa

Descripción generada automáticamente

Compiladores

Introducción

Mtro. Miguel Angel Romo Martínez

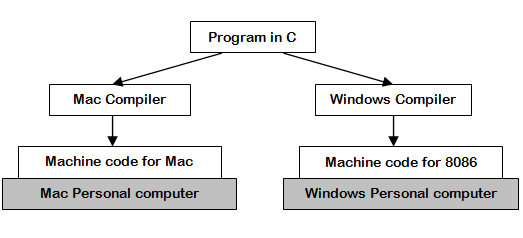
Enero – junio 2023

## Conceptos generales

**Formas de generar la ejecución de un programa:**

**Por Compilación**

*Ejemplos: C, C++*

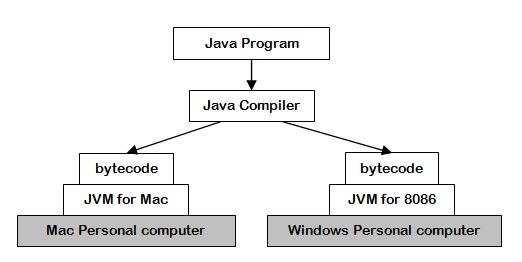


**Por Interpretación**

Se ejecuta línea por línea. Se puede pensar que el intérprete simula un procesador que en lugar de ejecutar instrucciones máquina, ejecuta instrucciones de alto nivel.

*Ejemplos:* Python y JavaScript

**Por Máquina virtual**



*Ejemplos:* Java y C#

## Ventajas y desventajas de lenguajes compilados e interpretados

1. Los lenguajes *compilados* suelen ser considerados de más bajo nivel que los interpretados. Suelen ser más rápidos en ejecución y consumir menos recursos, aunque son criticados por su falta de compatibilidad.
2. Los lenguajes *interpretados* suelen ser más potentes y flexibles, pero a costa de velocidad y consumo de recursos. Los lenguajes interpretados suelen permitir el uso de variable sin tipo definido lo que se suele ver como un incremento en la potencia y productividad, aunque puede tener gran costo produciendo errores difíciles de detectar.
3. Los lenguajes con *máquina virtual* son los que cuentan con más portabilidad, pero al ser interpretados en una máquina virtual su velocidad de ejecución es menor que interprete.

Para más información:

<http://www.edu4java.com/es/conceptos/compilador-interprete-maquina-virtual.html>

**Historia del los Compiladores**

1. Revisa con tu grupo el video: <https://youtu.be/TnsebcR_dTg>.
2. Con los años y sus respectivos sucesos genera una línea del tiempo:

|  |  |
| --- | --- |
| **Año** | **Sucesos** |
| 1950 | John Buck dirige una investigación en IBM en un lenguaje algebraico. |
| 1954 | Se comienza a desarrollar FORTRAN. |
| 1957 | FORTRAN se utiliza en la IBM modelo 704. Surge el concepto del traductor. |
|  | Se crea el compilador 18 años después |
| 1957 | FORTRAN se desarrolla en América “Gramáticas Libres de Contexto”(se pretenden lenguajes independientes) |
| 1958 | Surge un grupo europeo encabezado por la FL. Lenguaje internacional. Se propone la solución de que un compilador fuera portable por lo que se divide en dos fases:   * **Front-end:** Analiza el programa fuente * **Back-end:** Genera código objeto para la máquina objeto   Lenguaje intermedio “UNCOL”. Pero no funcionó :c |
| 1959 | Robin y Scott proponen   * **AFD**: Autómatas finitos deterministas * **AFN**: Autómatas finitos NO deterministas   Se eliminan ambigüedades.  Método de parsing de FORTRAN para introducir paréntesis en una expresión. |
| 1960 | Técnica de precedencia de operadores y uso de funciones de precedencia.  Aparece Backus Naur Form (BNF). Para el Desarrollo del análisis sintáctico. |
| 1961 | Se usa por primera vez un parser recursivo. Se incluye el pase de parámetros por nombre, valor y referencia. Uso dinámico de datos. |
| 1968 | Se estudia y se definen las GLC (Gramática Libre de Contexto). Los parses predictivos y la eliminación de recursividad izquierda. |
| 1975 | Aparece el generador automático de analizadores léxicos a partir de expresiones regulares bajo UNIX.  Johnos crea Yacc como generador de analizadores sintácticos. Crea el lenguaje **C** basado en **B.** |
| 1980 | Se crea lenguaje ADA. |
| 1987 | Se crea el lenguaje Perl para procesamiento en sistemas unix. |
| 1991 | Python se crea. Se crean los hilos de programación para que se pusiera divertido el asunto. |
| 1993 | Ruby nace (Diseñado para la programación productiva y agradable). |
| 1995 | Se crea Java para un proyecto de televisión interactiva. Es interpretado. |
| 2009 | Se crea Go. Inspirado en Python. Lenguaje compilado y concurrente. C++ |

**Contesta el siguiente cuestionario.**

* + - 1. ¿Qué es un compilador**? Es un traductor que tiene como finalidad transformar un programa con un lenguaje de programación de alto nivel.**
      2. ¿Por qué surgen los ensambladores? **Porque antes era muy complicado escribir con 1s y 0s.**
      3. ¿Cuál es problema de que un traductor dependa del equipo? **Se vuelve más complicado por ser para ciertas computadoras.**
      4. ¿Qué mecanismo se fueron desarrollarrollando para lograr compiladores portables?
* **Front-end:** Analiza el programa fuente
* **Back-end:** Genera código objeto para la máquina objeto
  + - 1. ¿En qué año(s) se desarrollaron los lenguajes C++, Java y Python?
         1. **C++ --- 1979**
         2. **Java --- 1995**
         3. **Python --- 1991**

Sube al Moodle en un PDF, la línea del tiempo y las respuestas a las preguntas.

## Repaso de Frames en NetBeans

¿Para que se usan las instrucciones?

Float.parseFloat

Float.toString

## Actividad

De manera individual, elabora una calculadora similar a la siguiente:

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza baja

Puedes utilizar el siguiente tutorial para saber cómo se trabaja: <https://youtu.be/b7rC0BENpQw>

Personaliza tu calculadora (haciendo cambios de colores) y agrega los siguientes botones: **PI** (Math.PI), **cambio de signo** (+/-) y la **inversa** (1/x)

Sube al Moodle en un PDF, tres ejemplos de uso de la calculadora.