

UNIVERSIDAD PANAMERICANA

Compiladores

Introducción

Mtro. Miguel Angel Romo Martínez

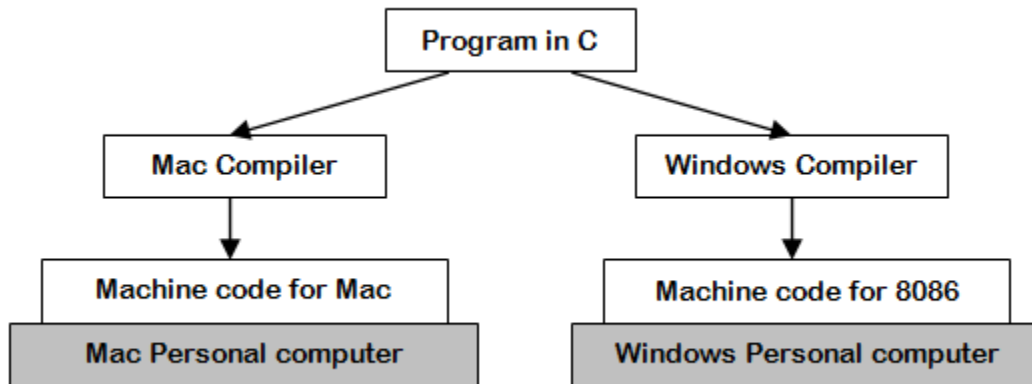
Enero – junio 2023

Conceptos generales

Formas de generar la ejecución de un programa:

Por Compilación

Ejemplos: C, C++

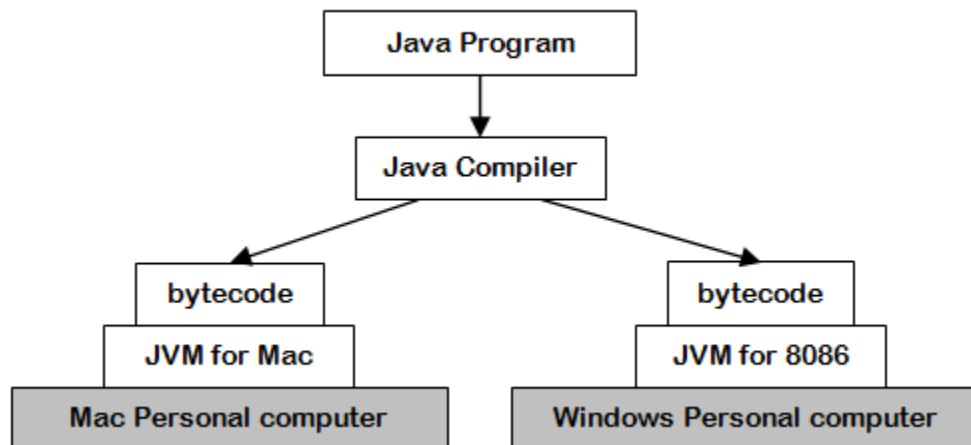


Por Interpretación

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

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

Año	Sucesos
1950	John Buck dirige una investigación en IBM para un lenguaje.
1954	Comienza el desarrollo de 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 para Gramáticas Libres de Contexto para hacer 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: <ul style="list-style-type: none">● Front-end: Analiza el programa fuente.● Back-end: Genera código objeto para la máquina objeto Lenguaje intermedio "UNCOL". Pero no funciona
1959	Robin y Scott propone: <ul style="list-style-type: none">● AFD: Autómatas finitos deterministas.● AFN: Autómatas finitos no deterministas. Se eliminan ambigüedades. Método de Parsing para introducir paréntesis de una expresión.
1960	Floyd introduce la técnica de operadores y funciones de precedencia. Aparece BNF (Backus Naur Form) como una guía para análisis sintáctico.

1961	Se usa por primera vez un parser recursivo con parámetros por nombre, valor y referencia. También entra el uso dinámico de datos.
1968	Se estudia y se define la gramática libre de contexto los parsers predictivos y la eliminación de recursividad izquierda.
1975	Aparece <i>lex</i> un generador automático de generadores léxicos a partir de expresiones regulares bajo UNIX. Surge <i>Yacc</i> como generador de analizadores sintácticos. Surge C basado en B. Sus derivados C#, Java, JS, Perl, PHP, python Ruby, Objective-C, Swift.
1979	Surge C++.
1980	Nace el lenguaje ADA.
1987	Aparece PERL para procesar informes en UNIX.
1991	Aparece python para apoyar muchos estilos de programación, como una alternativa divertida.
1993	Aparece ruby para la programación productiva y agradable.
1995	Surge Java para un proyecto de televisión interactiva.
2009	Aparece Golang que permite la concurrencia

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 a lenguaje maquina.**
2. ¿Por qué surgen los ensambladores? **Porque antes era muy complicado escribir en lenguaje binario.**
3. ¿Cuál es el problema de que un traductor dependa del equipo? **Sirve solo para computadores en específico y dificulta el progreso de la tecnología.**
4. ¿Qué mecanismo se fueron desarrollaron para lograr compiladores portables? **Dividirlo en dos etapas.**
5. ¿En qué año(s) se desarrollaron los lenguajes C++, Java y Python? **1979, 1995, 1991 respectivamente.**

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

Repaso de Frames en NetBeans

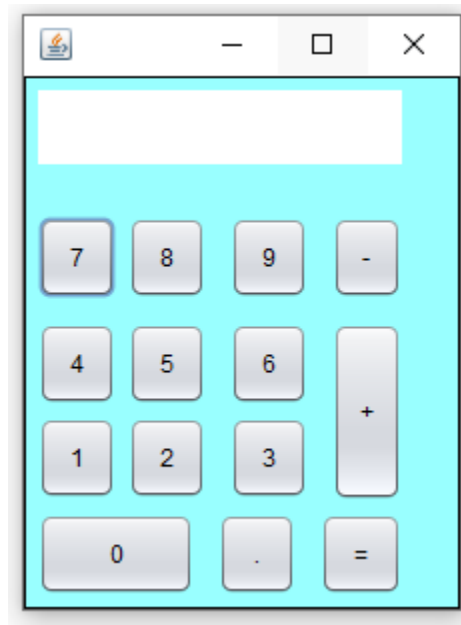
¿Para que se usan las instrucciones?

`Float.parseFloat`

`Float.toString`

Actividad

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



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.