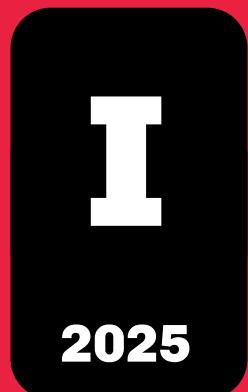


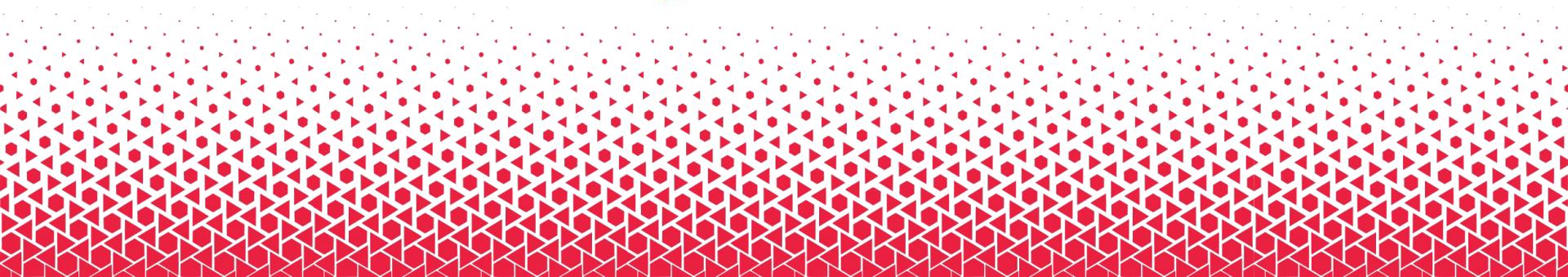
# ¿Qué es Java?

**UNRN**

Universidad Nacional  
de Río Negro

rXXI





**Creado en ~1994  
Por James Gosling**

**En Sun Microsystems**



---

# ¿Qué es?

# Java es:

**1**

**Es un lenguaje de  
programación  
Orientado a Objetos**

**2**

# **De tipado fuerte (como C)**

3

**Es compilado**

**4**

**No corre  
directamente en la  
computadora**

**5**

**Corre sobre una  
“plataforma”**

**En constante  
actualización  
(¡cada 6 meses!)**

**Vamos a usar la  
versión 21\***

A yellow cube with two large white question marks on its faces, resembling a power-up item from the Super Mario video game series.

**¿Preguntas?**



# ¿Cómo se ejecuta un programa?

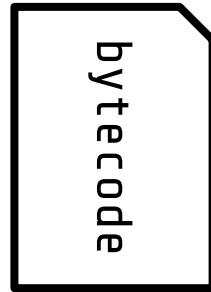
a grandes rasgos

**Esto es lo  
que uno ve**



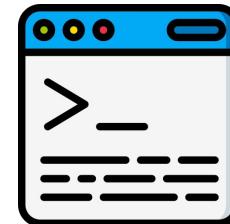
Programa.java

javac

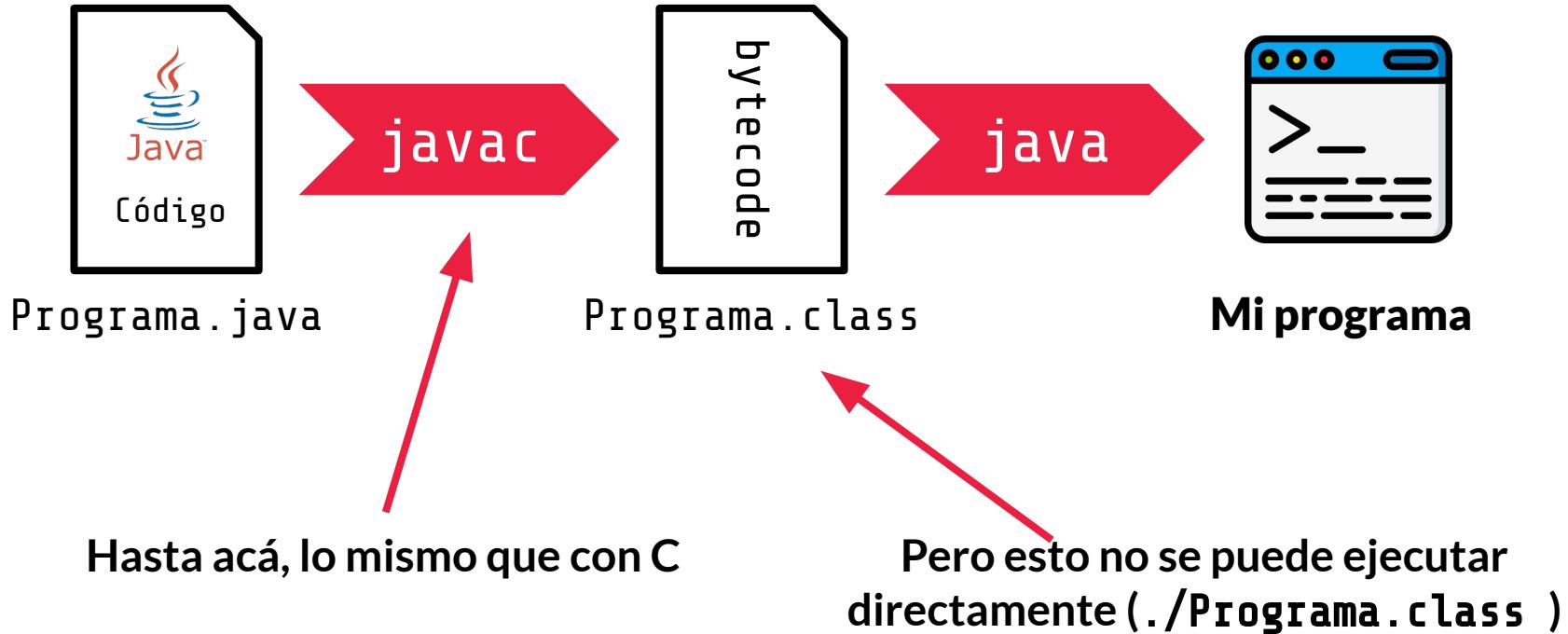


Programa.class

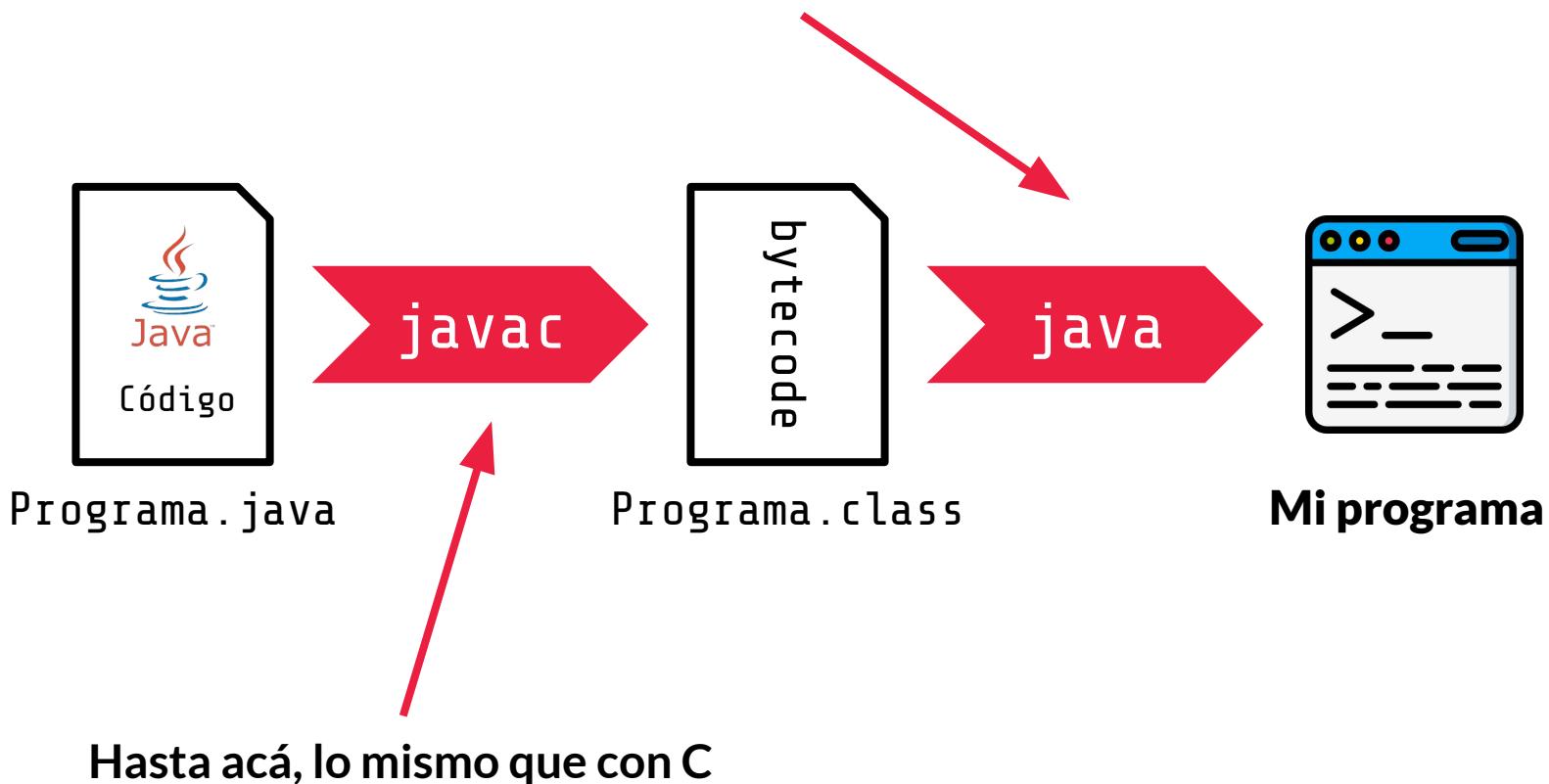
java



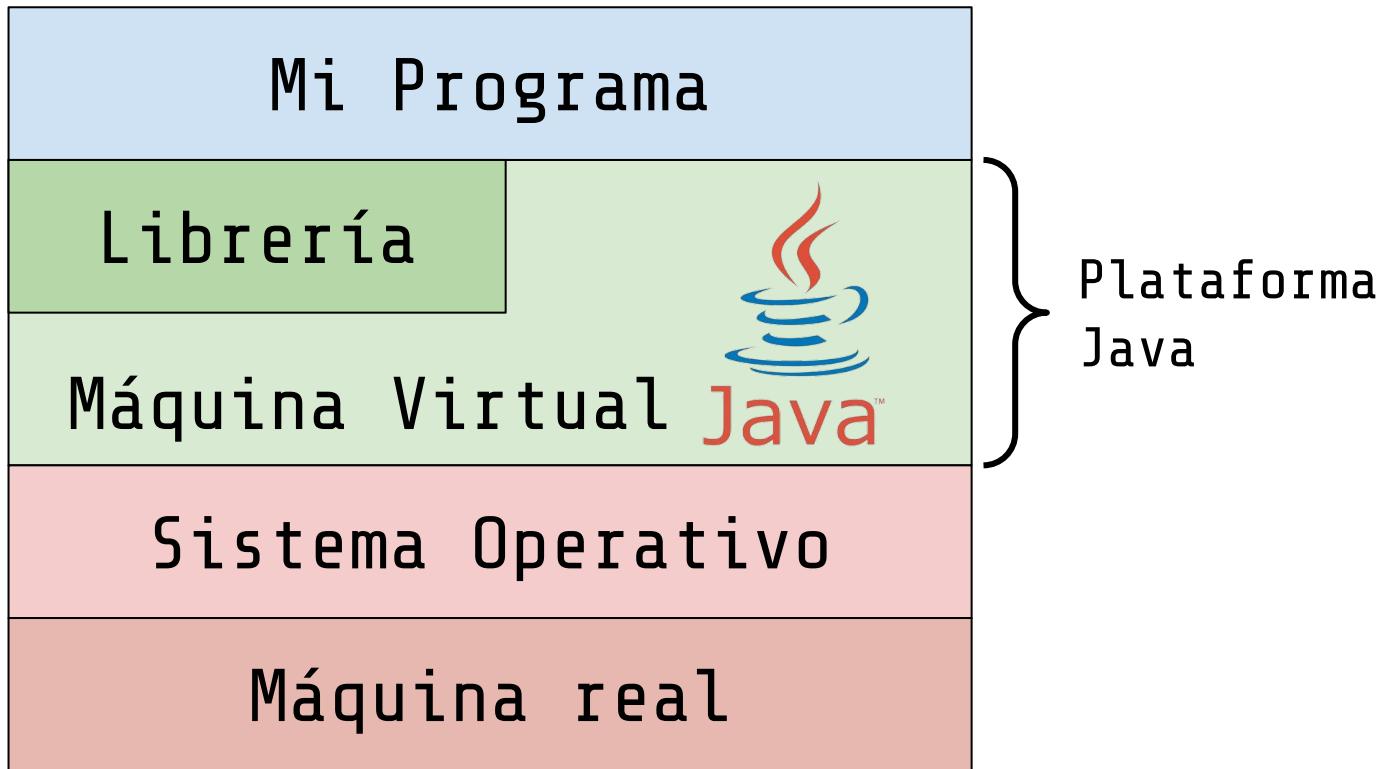
**Mi programa**

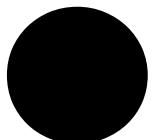


**Se ejecuta como java Programa**



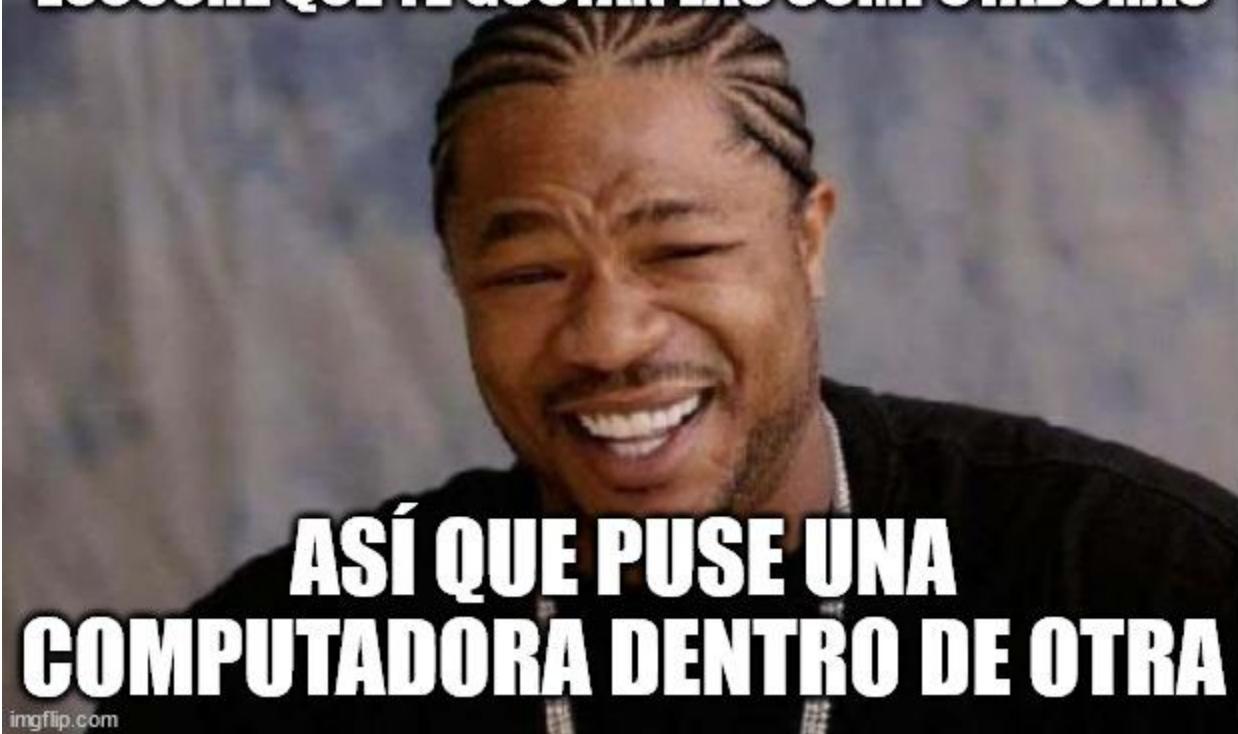
**La plataforma está compuesta de:**  
**La máquina virtual**  
**Librerías de soporte**





# ¿Máquina virtual?

**ESCUCHÉ QUE TE GUSTAN LAS COMPUTADORAS**



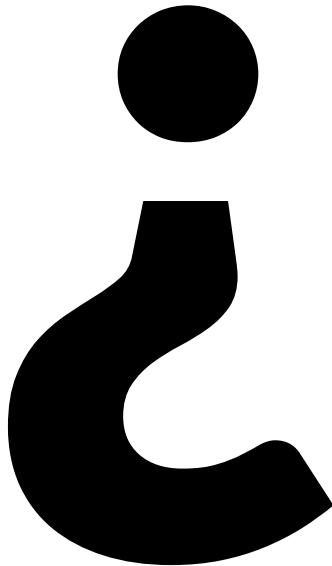
**ASÍ QUE PUSE UNA  
COMPUTADORA DENTRO DE OTRA**

**O**  
**un programa que  
actúa como una  
computadora**

**La llamaremos**

**JVM**

**Java Virtual Machine**

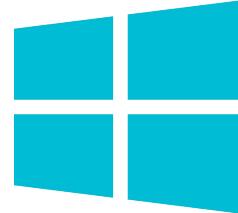


Pero...  
Por qué



**Trae muy  
interesantes  
ventajas**

**Un .class funciona  
en cualquier JVM  
sin recompilar**



**Esto es ‘portabilidad’**

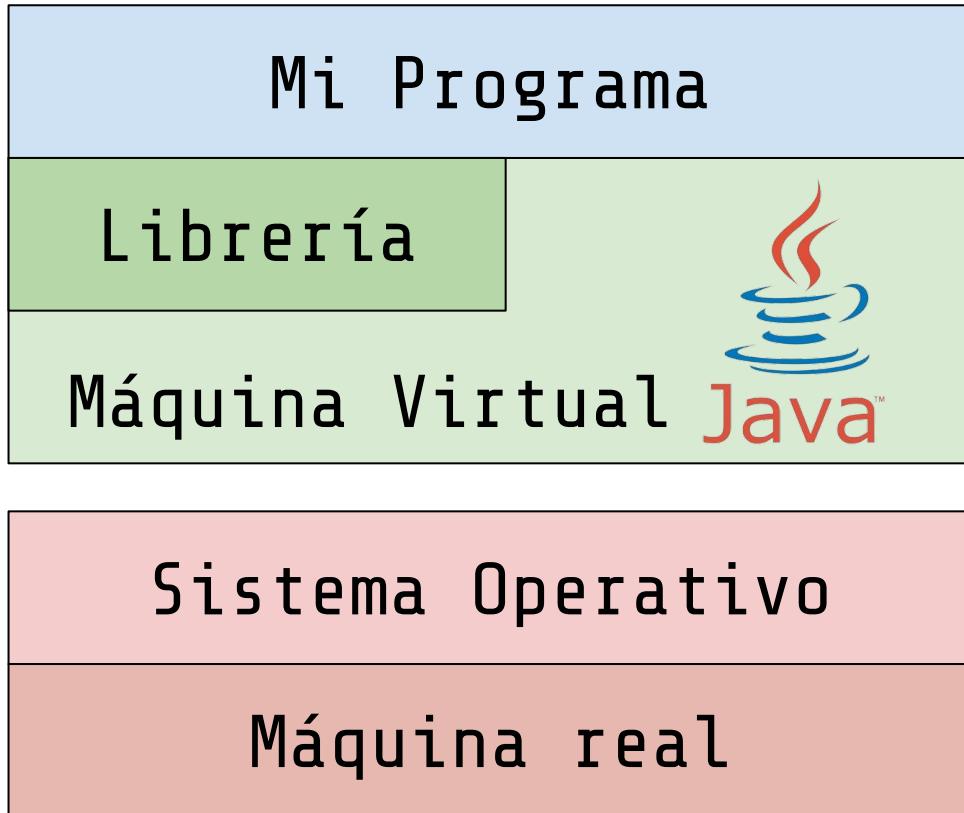
arm

intel®

**Independiente del  
procesador real**

 RISC-V®

**Ademas del sistema operativo**



Nos independiza de lo que  
este abajo

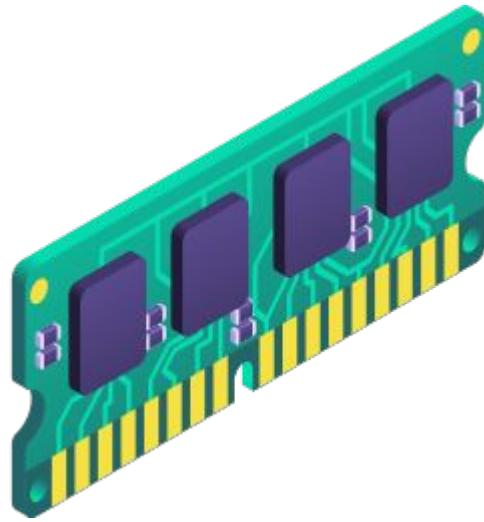
# Gestión de errores detallada



# Seguridad



# **La gestión de memoria es automática**





**La JVM puede  
ejecutar código  
en otros  
lenguajes**

**JS**



*Como programa,*  
podemos analizar la  
JVM en gran detalle

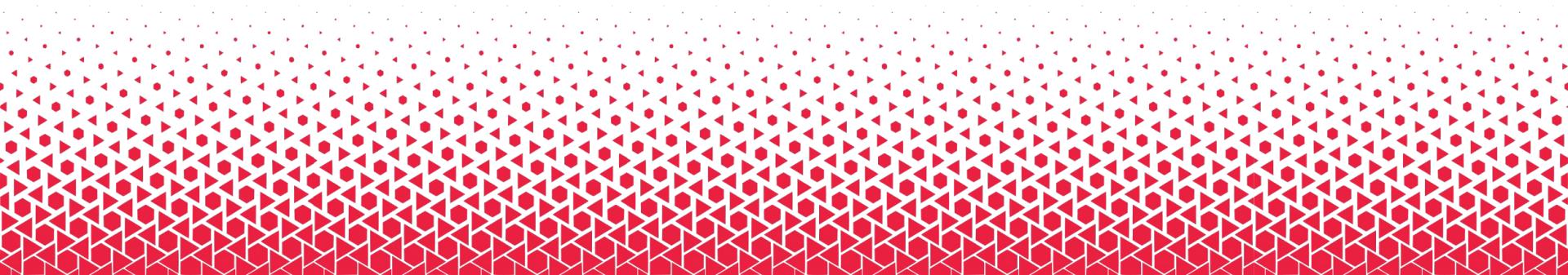


# De hecho

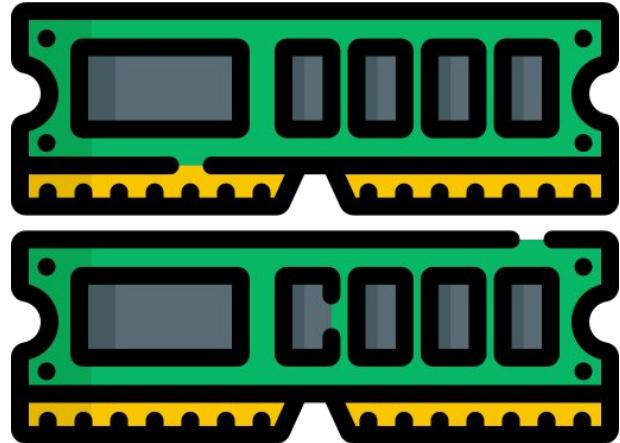
La JVM utiliza una  
técnica llamada JIT

**Que compila partes  
del programa a  
código de máquina.**

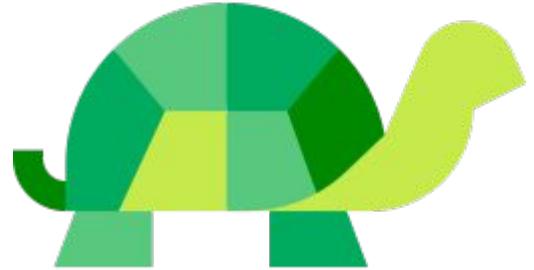
# Pero



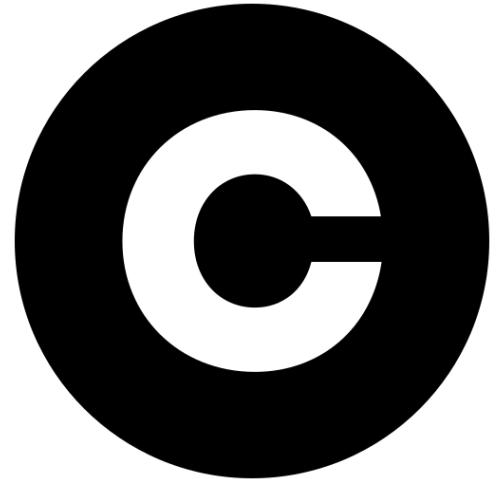
# Requiere más memoria



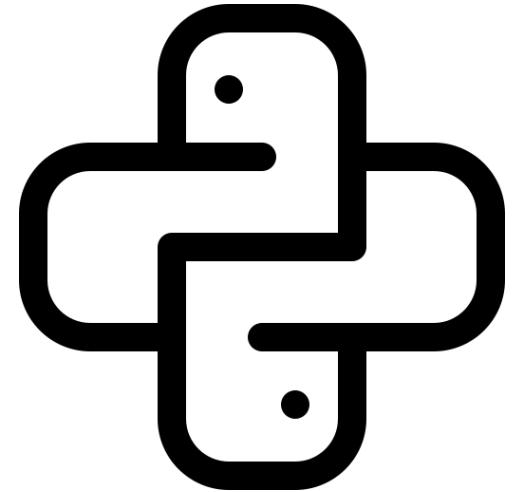
# Son de ‘arranque lento’



**Es más lento que  
un programa en C\***



**Es bastante más  
rápido que uno en  
Python**



**La plataforma es  
grande y compleja**





**¿Preguntas?**



# Hasta la próxima



**unrn.edu.ar**

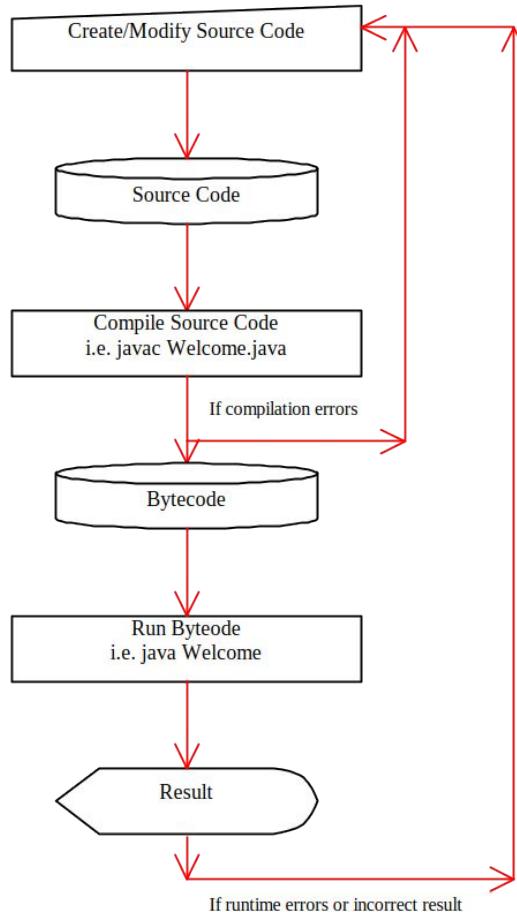
**UNRN**

Universidad Nacional  
de Río Negro

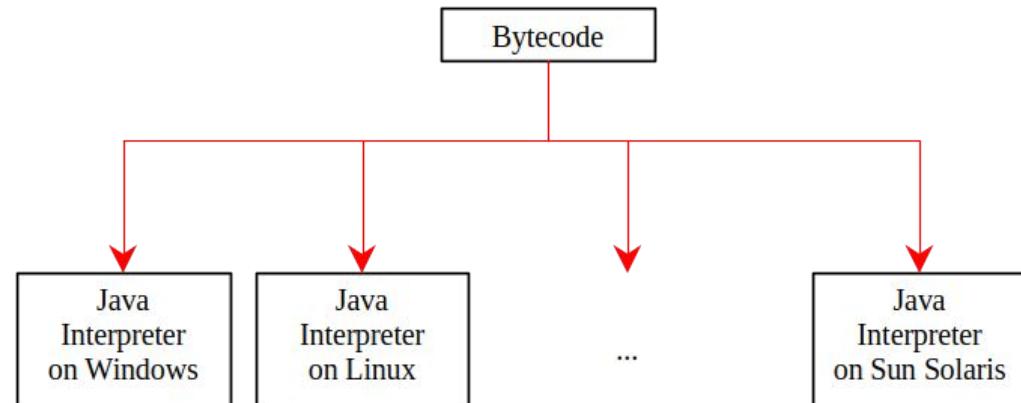


| unrnionegro

# Executing Applications



- On command line
  - **java classname**



Devices		
Access control systems	Airplane systems	ATMs
Automobiles	Blu-ray Disc™ players	Building controls
Cable boxes	Copiers	Credit cards
CT scanners	Desktop computers	e-Readers
Game consoles	GPS navigation systems	Home appliances
Home security systems	Internet-of-Things gateways	Light switches
Logic controllers	Lottery systems	Medical devices
Mobile phones	MRIs	Network switches
Optical sensors	Parking meters	Personal computers
Point-of-sale terminals	Printers	Robots
Routers	Servers	Smart cards
Smart meters	Smartpens	Smartphones
Tablets	Televisions	Thermostats
Transportation passes	TV set-top boxes	Vehicle diagnostic systems

**Fig. 1.1** | Some devices that use Java.



## Software Engineering Observation 1.1

Use a building-block approach to creating your programs. Avoid reinventing the wheel—use existing high-quality pieces wherever possible. This software reuse is a key benefit of object-oriented programming.

# Scoring System "Java Fundamentals" Course

- Exam – 80%
- Labs – 10% (added to the exam results)
- Homework + evaluation – 5% + 5%
- Team work – 10%
- Bonuses – up to 10%
- Presence in class – 5% (onsite students only)

