

# Compiladores e Interpretadores

Rios Quijada Danira

15 de Febrero de 2015

## 1. Introducción

Un compilador analiza el programa y lo traduce al idioma "maquina". La acción fundamental los compiladores es equivalente a la de un traductor humano, que toma nota de lo que esta escuchando y reproduce por escrito en otra lengua. Por otro lado los interpretadores analizan el programa fuente y lo ejecutan directamente, o sea en el ejemplo del traductor humano, éste sería un traductor humano que conforme a lo que está escuchando va ejecutando, sin generar ningún escrito, es decir que sobre la marcha va traduciendo.

## 2. Tabla comparativa entre distintos lenguajes de programación

Nombre	Paradigma	Creadores	Año	Extensión
C	Imperativo	Dennis M. Ritchie	1972	.c
C++	Imperativo, orientado a objetos	Bjarne Stroustrup	1983	.cpp
Java	Orientado a objetos, imperativo	Sun Microsystems	1995	.java
Fortran	Imperativo	IBM	1957	.f90 (Depende/versión)
Python	Funcional, reflexivo, O.O	Guido van Rossum	1991	.py
Ruby	O.O, reflexivo	Yukihiro Matsumoto	1995	.rb

## 3. Ejemplo del programa Adivina el resultado de las operaciones mentales al haber escogido un número previamente y de común acuerdo con el usuario, en cada uno de los lenguajes descritos

### 3.1. C

```
#include <stdio.h>
#include <unistd.h>
```

```

int main()
{
    printf("Hola! Trataré de adivinar un número.\n");

    printf("Piensa un número entre 1 y 10.\n");
    sleep(5);
    printf("Ahora multiplícalo por 9.\n");
    sleep(5);
    printf("Si el número tiene 2 dígitos, súmalos entre si: Ej. 36 -> 3+6=9. Si tu número t");
    sleep(5);
    printf("Al número resultante súmale 4.\n");
    sleep(10);
    printf("Muy bien. El resultado es 13 :) \n");

}

```

### 3.2. C++

```

#include <iostream>
#include <unistd.h>

int main()
{
    std::cout << "Hola! Trataré de adivinar un número.\n";
    std::cout << "Piensa en un número entre 1 y 10\n";
    sleep(5);
    std::cout << "Ahora multiplícalo por 9.\n";

    sleep(5);
    std::cout << "Si el número tiene 2 dígitos, súmalos entre si: Ej. 36 -> 3+6=9. Si

    sleep(5);
    std::cout << "Al número resultante súmale 4.\n";

    sleep(10);
    std::cout << "Muy bien. El resultado es 13 :D\n";

    return(0);
}

```

### 3.3. Java

```

// Hola mundo en Java
class Holamundo {
static public void main( String args[] ) {

```

```

System.out.println("Hola! Trataré de adivinar un número.");
System.out.println("Piensa un número entre 1 y 10.");
try {
    Thread.sleep(4000);
} catch (InterruptedException ex) {
    Thread.currentThread().interrupt();
}
System.out.println("Ahora multiplícalo por 9.");
try {
    Thread.sleep(4000);
} catch (InterruptedException ex) {
    Thread.currentThread().interrupt();
}
System.out.println("Si el número tiene 2 dígitos, súmalos entre si: Ej. 36 -> 3+6=9. Si tu n
try {
    Thread.sleep(4000);
} catch (InterruptedException ex) {
    Thread.currentThread().interrupt();
}
System.out.println("Al número resultante súmale 4.");
try {
    Thread.sleep(8000);
} catch (InterruptedException ex) {
    Thread.currentThread().interrupt();
}
System.out.println("Muy bien. El resultado es 13 [:");
}

```

### 3.4. Fortran

```

program hola
write(*,*) 'Hola! Trataré de adivinar un número.'
write(*,*) 'Piensa un número entre 1 y 10.'
call sleep(5)
write(*,*) 'Ahora multiplícalo por 9.'
call sleep(5)
write(*,*) 'Si el número tiene 2 dígitos, súmalos entre si: Ej. 36 -> 3+6=9. Si tu número t
call sleep(5)
write(*,*) 'Al número resultante súmale 4.'
call sleep(10)
write(*,*) 'Muy bien. El resultado es 13 n.n'
end program hola

```

### 3.5. Python

```

# Hola mundo en Python

```

```
# -*- coding: 850 -*-
import time
print "Hola! Trataré de adivinar un número."
print "Piensa un número entre 1 y 10."
import time
time.sleep(5)
print "Ahora multiplícalo por 9."
import time
time.sleep(5)
print "Si el número tiene 2 dígitos, súmalos entre si: Ej. 36 -> 3+6=9. Si tu número tiene u
import time
time.sleep(5)
print "Al número resultante súmale 4."
import time
time.sleep(10)
print "Muy bien. El resultado es 13 :P "
```

### 3.6. Ruby

```
# -*- coding: utf-8 -*-
# Hola mundo en Ruby
#encoding: utf-8
puts "Hola! Trataré de adivinar un número."
puts "Piensa un número entre 1 y 10."
sleep(5)
puts "Ahora multiplícalo por 9."
sleep(5)
puts "Si el número tiene 2 dígitos, súmalos entre si: Ej. 36 -> 3+6=9. Si tu número tiene un
sleep(5)
puts "Al número resultante súmale 4."
sleep(10)
puts "Muy bien. El resultado es 13 :V"
```