

Compiladores e Interpretadores

Luisa Fernanda Orci Fernandez.

15 de Febrero de 2015

1. Introducción

Un compilador es un programa informático que traduce un escrito de programación a un lenguaje de programación diferente, generando un programa que otra máquina pueda interpretar.

Un interpretador o intérprete, es un programa informático que puede analizar y ejecutar otros programas. La diferencia al compilador es que el interpretador solo realiza la traducción instrucción tras instrucción y no guarda el resultado de la traducción.

2. Tabla comparativa entre algunos 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. A continuación se muestran ejemplos de estos lenguajes de programación; mediante el juego “Adivina el número”

3.1. C

```
/* Juego en C, Ansi-Style */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
```

```

int main(void)
{
    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 dígito, multiplícalo por 9.");
    sleep(5);
    puts("Al número resultante súmale 4.");
    sleep(8);
    puts("Muy bien. El resultado es 13 :)");
    return(EXIT_SUCCESS);
}

```

3.2. C++

```

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

int main()

{
    std::cout << "Hola! Trataré de adivinar un número. 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 tu número tiene un dígito, multiplícalo por 9.\n";
    sleep(5);
    std::cout << "Al número resultante súmale 4.\n";
    sleep(8);
    std::cout << "Muy bien. El resultado es 13 :)\n";
    return(0);
}

```

3.3. Fortran90

```

program juego
    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 tiene un dígito, multiplícalo por 9.'
end

```

```

    call sleep(5)
    write (*,*) 'Al número resultante súmale 4.'
    call sleep(8)
    write (*,*) 'Muy bien. El resultado es 13 :)'
end program juego

```

3.4. Java

```

// Juego en Java
class juego {
    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(5000);
    } catch(InterruptedException ex) {
Thread.currentThread().interrupt();
    }
System.out.println("Ahora multiplícalo por 9.");
    try {
Thread.sleep(5000);
    } 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(5000);
    } 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 :)");
    }
}

```

3.5. Python

```

# Juego 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(8)
print "Muy bien. El resultado es 13 :)."
```

3.6. Ruby

```

# -*- coding: utf-8 -*-
# Juego 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(8)
puts "Muy bien. El resultado es 13 :)."
```