DPL

(P1)

Yeray Méndez Romero

[yeray.mendez@udc.es](mailto:yeray.mendez@udc.es)

Daniel Rivera López

[d.rivera1@udc.es](mailto:d.rivera1@udc.es)

Lenguajes seleccionados:

-oCaml

-Python

-Go

-Kotlin

-Groovy

-lua

oCaml:

a)

-Compilador: version 4.02.3

-Sistema: Ubuntu 16.04 LTS

b)

(preguntar compilar, interpretar)

OCaml es un fuerte lenguaje industrial de programación que soporta el paradigma funcional, imperativo y la orientación a objetos y tiene un sistema automático de gestión de memoria (recolector de basura). El código fuente de OCaml se compila en código para máquina virtual y en código máquina para diferentes arquitecturas, en el último caso con una eficiencia comparable a C/C++. OCaml también dispone de un análisis de tipos estáticos con inferencia de tipos, con un sistema de tipos fuerte, con valores funcionales de primera clase, polimorfismo parametrizado, llamada por patrones, manejo de excepciones y otras características avanzadas.

(Preguntar duda paradigma)

c)

\*Ventajas:

-Tipos definibles por el usuario lo que nos permite mantenernos más fieles a la implementación original y evitar recurrir a la OO.

-The memory management is done automatically by the OCaml which release us from explicit management of the memory.

-Gestión explicita de punteros, ya que nos permite mantenernos fieles a la implementación original.

-Permite programación imperativa, lo que nos permite mantenernos fieles a la implementación original.

\*Desventajas:

d)

-Dado que las funciones de OCaml deben devolver siempre un dato de algún tipo, fue necesario que este tipo fuera unit para mantener la esencia de la implementación original

-(error????????)

e)

-Los métodos iterativos nos obligan a perder el potencial recursivo de OCaml.

Python:

a)

-Interpreter: Python 3.4.4 – Python 3.5.2

-System: Windows 10 – Ubuntu 16.04

b)

Python is a simple and easy to learn high level programming language which supports multiple programming paradigms including Object-Oriented, imperative, functional programming and procedural styles. Python also features a dynamic and strong typing and automatic memory management. Python is an interpreted programming language which makes python programs much more portable than a compiled language. Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords)

c)

\*Advantages:

-Python allows imperative programming which make us able to do the iterative methods.

-The memory management is done automatically by the Python memory manager which release us from explicit management of the memory.

\*Disadvantages:

-Python does not allow the creation of custom data types without importing a library and because of that the custom type is made by objects.

d)

-As we previously said we were forced to use a class and objects from it to mimic the custom type of Pascal, otherwise we would have needed to import a library.

e)

-(Objetos clase cosa)