

Estudo de idoneidade de diferentes linguaxes de programación na aprendizaxe en Programación do Grao de Enxeñaría Informática

Castro Martínez, Alfonso; Arcay Varela, Bernardino; Bóveda Álvarez, María del Carmen

*Grupo de Innovación Educativa en Programación, Facultade de Informática,
Universidade de A Coruña*

PALABRAS CLAVE: linguaxes de programación, docencia en grao, recursos tecnolóxicos na aprendizaxe, melloras docentes.

RESUMO

A materia de Programación forma parte do módulo de formación básica do Grao en Informática, o obxectivo desta materia é ensinar ao alumno a crear modelos computacionais de problemas do mundo real, a realización desta labor faise empregando unha linguaxe de programación.

A linguaxe de programación empregada na materia para ensinar os alumnos é un elemento fundamental e crítico, xa que dependendo das súas características vai ser máis doado ou complicado acadar as competencias para o alumno: detección de erros, escritura dos programas, entorno de desenvolvemento (IDE), ... Unha mostra disto é que se emprega a mesma linguaxe desde a creación da Facultade de Informática (FIC).

Está non é un tarefa trivial debido a que non tódalas linguaxes están deseñadas seguindo o mesmo paradigma de programación, algunhas son case experimentais mentres outras son de uso case exclusivamente comercial, hai linguaxes que non teñen compilador “propio”, ...

Neste traballo se presentan o resultado da análise de diferentes linguaxes que compren coas características necesarias, como se indicou anteriormente, para o seu uso na docencia da materia, valorando a súa idoneidade nas características máis importantes desde un punto de vista docente. As linguaxes que observouse que as cumprían foron: C/C++, C#, Python, Java, Ada, Go, Dart, Pascal, Fortran, Haskell, Scala, Nim, D e Kotlin.

III Xornadas de innovación docente da UDC

“Contextos universitarios transformadores: construíndo espazos de aprendizaxe”

8 e 9 novembro de 2018

Estudo de idoneidade de diferentes linguaxes de programación na aprendizaxe en Programación do Grao de Enxeñaría Informática

Contextualización

A materia de Programación forma parte do módulo de Formación Básica do primeiro curso do Grao en Informática, o número de créditos ECTS é 6, impárase no primeiro cuadrimestre, o obxectivo desta materia é ensinar o alumno a crear modelos computacionais de problemas do mundo real, os cales se implementan nun algoritmo computacional, a realización desta labor faise empregando unha linguaxe de programación.

Nesta materia houbo 301 alumnos matriculados no curso 2017/2018 e divídese en 6 grupos de teoría e 21 grupos de prácticas, impartíndose en: castelán, galego e inglés. A docencia distribúese en 2 horas de clase de teoría e 2 de práctica semanalmente por grupo. O núcleo da materia son as prácticas xa que só desenvolvendo programas o alumno poderá adquirir as competencias, en Informática dise: “a programar aprendese programando”

ID	Alumno	Asignatura	Materia	Alumno/Alta	ECTS	Símbolo	Unidade (semestre)
1.	Pr04	Programación I	Linguaxe e Programación	Informática	6	F04, F05	02
2.	AB	Informática Básica	Fundamentos de Informática	Informática	6	F04, F05	02
3.	GA	Cálculo	Matemáticas	Matemáticas	6	F04, F05	02
4.	MD	Matemática Discreta	Matemáticas	Matemáticas	6	F03, 02	02
5.	TE	Tecnoloxía Electrónica	Tecnoloxía, Estructura e Ingeniería de Computadores	Física	6	F02, 02	02
6.	Pr02	Programación II	Linguaxe e Programación	Informática	6	F02, F04	02
7.	IC	Fundamentos de los Computadores	Tecnoloxía, Estructura e Ingeniería de Computadores	Informática	6	F05, 02	02
8.	EE	Estadística	Matemáticas	Matemáticas	6	F01, 02	02
9.	AGO	Administración y Gestión de Organizaciones	Empresas	Empresas	6	F06, 02	02
10.	Alg	Algebra	Matemáticas	Matemáticas	6	F01, 02	02



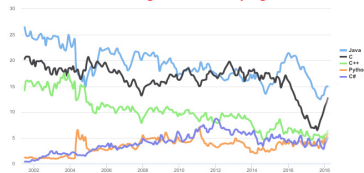
Características que debe ter unha linguaxe de programación para a docencia

- Paradigmas de programación: Enténdese por paradigma de programación un modelo básico de deseño e desenvolvemento de programas, que permite producir programas con unhas directrices específicas. Principalmente, están constituídos por unha serie de principios e normas que determinan o deseño, estrutura e construción dos programas. Cantos máis paradigmas soporte máis axeitada para a docencia é.
- Flexibilidade: as posibles formas de estrutura o código que permite a linguaxe. En linguaxes empregadas na docencia interesa o uso de linguaxes que sexan o menos ríxidas posibles, soportando o maior número de estruturas, proporcionando unha elevada flexibilidade na organización da teoría e as prácticas.
- Multiplataforma: que haxa ferramentas para poder traballar coa linguaxe no maior número de plataformas.
- Contorna de desenvolvemento (IDE): dispoñible de contornas de desenvolvemento de calidade profesional e dispoñible para o maior número de plataformas.
- Uso docente i empresarial: que sexa empregada por o maior número de entidades docentes e profesionais de prestixio. Contribuirá a un facilitar a incrementar o número de egresados.
- Documentación de apoio: ampla de dispoñibilidade de material de apoio (libros, tuto raís, exemplos,...) da linguaxe.

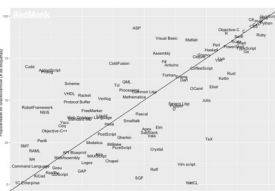
Linguaxes de programación analizadas axeitadas na docencia

C++	Java	Python	Scala	Haskell	Ruby	JavaScript	Dart						
<pre>#include <stdio.h> int main(void) { printf("Ola mundo!\n"); }</pre>	<pre>using System; namespace OlaMundo { class Ola { static void Main() { Console.WriteLine("Ola mundo!"); } } }</pre>	<pre>print("Ola mundo!")</pre>	<pre>public class OlaMundo { public static void main(String[] args) { System.out.println("Ola mundo!"); } }</pre>	<pre>with Ada.Text_IO; procedure Ola is begin Ada.Text_IO.Put_Line("Ola mundo!"); end Ola;</pre>	<pre>package main import "fmt" func main() { fmt.Println("Ola mundo!") }</pre>	<pre>void main() { print("Ola mundo!"); }</pre>	<pre>int main(input,output); begin write("Ola mundo!"); end.</pre>	<pre>program ola_mundo print *, "Ola mundo!" end program ola_mundo</pre>	<pre>main = putStrLn "Ola mundo!"</pre>	<pre>object OlaMundo { def main(args: Array[String]): Unit = { println("Ola mundo") } }</pre>	<pre>echo("Ola mundo!")</pre>	<pre>void main() { import std.stdio; writeln("Ola mundo!"); }</pre>	<pre>fun main(args : Array[String]) { println("Ola mundo") }</pre>

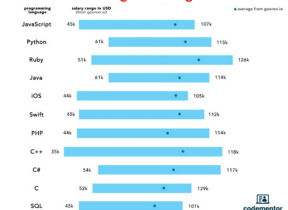
Evolución no uso das linguaxes máis empregadas



Linguaxes máis empregadas segundo Redmonk



Salarios en USA segundo a linguaxe de traballo



Linguaxes de programación máis axeitadas

C/C++

- Multiparadigma: soporta os paradigmas de referencia.
- A súa sintaxe é base doutras linguaxes: Java, Dart, ... O seu coñecemento facilitará a súa aprendizaxe.
- É moi empregada tanto na docencia como na industria.
- A tendencia da linguaxe é boa, sendo das linguaxes máis destacadas nos diferentes índices.
- Permite amosar o alumno unha versión xeral é xenérica dos diferentes aspectos de programación dunha computadora: tipo de datos, estruturas de control, manexo de memoria, ...
- É empregada na construción de pequenos e grandes sistemas informáticos.

Dart

- Multiparadigma: soporta os paradigmas de referencia.
- Concisa: non é no mesmo grado que Python pero suficiente para que sexa un punto a favor.
- Emprego na docencia e industria: aínda non é moi elevado para xa existe material e intención de fomentar o seu uso na docencia e na industria cada vez é maior.
- Boa tendencia: o seu uso como linguaxe de referencia en proxectos que se están consolidando fai que a expectativa do seu uso vaia a aumentar considerablemente nos próximos anos.

Alfonso Castro
Bernardino Arcay
Carmen Bóveda
alfonso.castro@udc.gal



Vicerreitoría de Oferta Académica e Innovación Docente
UNIVERSIDADE DA CORUÑA
482



Centro Universitario de Formación e Innovación Educativa