

Introducción

Ricardo Pérez López

IES Doñana, curso 2019/2020

Índice general

1. Conceptos básicos	2
1.1. Informática	2
1.2. Ordenador	2
1.3. Algoritmo	3
1.4. Programa	5
1.5. Lenguaje de programación	5
2. Evolución histórica	5
2.1. Culturas de la programación	5
2.2. Ingeniería del software	5
3. Resolución de problemas mediante programación	5
3.1. Análisis del problema	5
3.2. Especificación	5
3.3. Diseño del algoritmo	5
3.4. Codificación del algoritmo en forma de programa	5
4. Paradigmas de programación	5
4.1. Imperativo	5
4.2. Declarativo	5
5. Lenguajes de programación	5
5.1. Definición	5
5.2. Evolución histórica	6
5.3. Clasificación	6
6. Traductores	6
6.1. Compiladores	6
6.2. Interpretes	6
7. Entornos integrados de desarrollo	6
7.1. Terminal	6
7.2. Editores de texto	6

1. Conceptos básicos

1.1. Informática

- Definición:
La ciencia que estudia los sistemas de tratamiento automático de la información, también llamados **sistemas informáticos**.
- Estos sistemas están formados por elementos físicos, lógicos y humanos.
- A los elementos físicos se les denomina **hardware** y a los elementos lógicos se les denomina **software**.
- El *hardware*, a su vez, está formado por componentes:
 - Ordenadores
 - Redes de comunicaciones
 - Soportes de almacenamiento
 - ...

1.2. Ordenador

1.2.1. Definición

Un **ordenador** es una máquina que procesa información automáticamente de acuerdo con un programa almacenado.

1. Es una *máquina*.
2. Su función es *procesar información*.
3. El procesamiento se realiza de forma *automática*.
4. El procesamiento se realiza siguiendo un *programa*.
5. Este programa está *almacenado* en una memoria interna del mismo ordenador (arquitectura de **Von Neumann**).

1.2.2. Funcionamiento básico

xxx

1.2.2.1. Elementos funcionales xxxx

1.2.2.2. Ciclo de instrucción

1.2.2.3. Representación de información

1.2.2.3.1. Codificación interna Sistema binario

1.2.2.3.2. Codificación externa ASCII

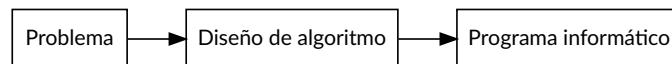
Unicode

1.3. Algoritmo

1.3.1. Definición

Un **algoritmo** es un método para resolver un problema.

- Está formado por una secuencia de pasos o **instrucciones** que se deben seguir (o **ejecutar**) para resolver el problema.
- La palabra «algoritmo» proviene de **Mohammed Al-Khowârizmi**, matemático persa que vivió durante el siglo IX y reconocido por definir una serie de reglas paso a paso para sumar, restar, multiplicar y dividir números decimales.
- **Euclides**, el gran matemático griego (del siglo IV a. C.) que inventó un método para encontrar el máximo común divisor de dos números, se considera con Al-Khowârizmi el otro gran padre de la Algorítmica (la ciencia que estudia los algoritmos).
- El estudio de los algoritmos es importante porque la resolución de un problema exige el diseño de un algoritmo que lo resuelva.



Resolución de un problema

1.3.2. Características

- Un algoritmo debe ser:
 - **Preciso**: debe indicar el orden de ejecución de cada paso.
 - **Definido**: si se sigue un algoritmo dos veces, se debe obtener el mismo resultado cada vez.
 - **Finito**: debe terminar en algún momento, es decir, debe tener un número finito de pasos.

1.3.3. Representación

1.3.3.1. Ordinograma

1.3.3.2. Pseudocódigo

1.3.4. Cualidades deseables

1.3.5. Computabilidad

1.3.6. Corrección

1.3.7. Complejidad

1.4. Programa

1.5. Lenguaje de programación

2. Evolución histórica

2.1. Culturas de la programación

2.2. Ingeniería del software

3. Resolución de problemas mediante programación

3.1. Análisis del problema

3.2. Especificación

3.3. Diseño del algoritmo

3.4. Codificación del algoritmo en forma de programa

4. Paradigmas de programación

4.1. Imperativo

4.1.1. Estructurado

4.1.2. Orientado a objetos

4.2. Declarativo

4.2.1. Funcional

4.2.2. Lógico

5. Lenguajes de programación 5

5.1. Definición

5.1.1. Sintaxis

5.1.1.1. Notación EBNF

5.1.2. Semántica

5.2. Evolución histórica

5.3. Clasificación

5.3.1. Por nivel

5.3.2. Por generación

5.3.3. Por paradigma

6. Traductores

6.1. Compiladores

6.2. Intérpretes

6.2.1. Interactivos (*REPL*)

6.2.2. Por lotes

7. Entornos integrados de desarrollo

7.1. Terminal

7.1.1. [Zsh](#)

7.1.2. Oh My Zsh

7.1.3. [less](#)

7.2. Editores de texto

7.2.1. Editores vs. IDE

7.2.2. Vim y less

7.2.3. Visual Studio Code

7.2.3.1. Instalación

7.2.3.2. Configuración

7.2.3.3. Extensiones