Bloque 1. Conceptos y técnicas básicas en programación

Notas:

1. Introducción

- Programas, y lenguajes. Necesidad de la programación modular. Ciclo de vida de los programas. Concepto de algoritmo. Noción de proceso. Variables. Estado de un programa.

2. Datos y expresiones. Especificación de algoritmos

3. Estructuras algorítmicas básicas

4. Iteración y recursión

5. Iteración y recursión sobre secuencias

6. Iteración y recursión sobre tablas

Programas y lenguajes

Un programa es un conjunto completo de instrucciones que el computador es capaz de ejecutar con el fin de tratar información

Se escribe en un lenguaje que el computador entienda

- **lenguaje de programación**
- **generalmente es necesario un proceso de traducción automática (compilación)**

Necesidad de la programación modular

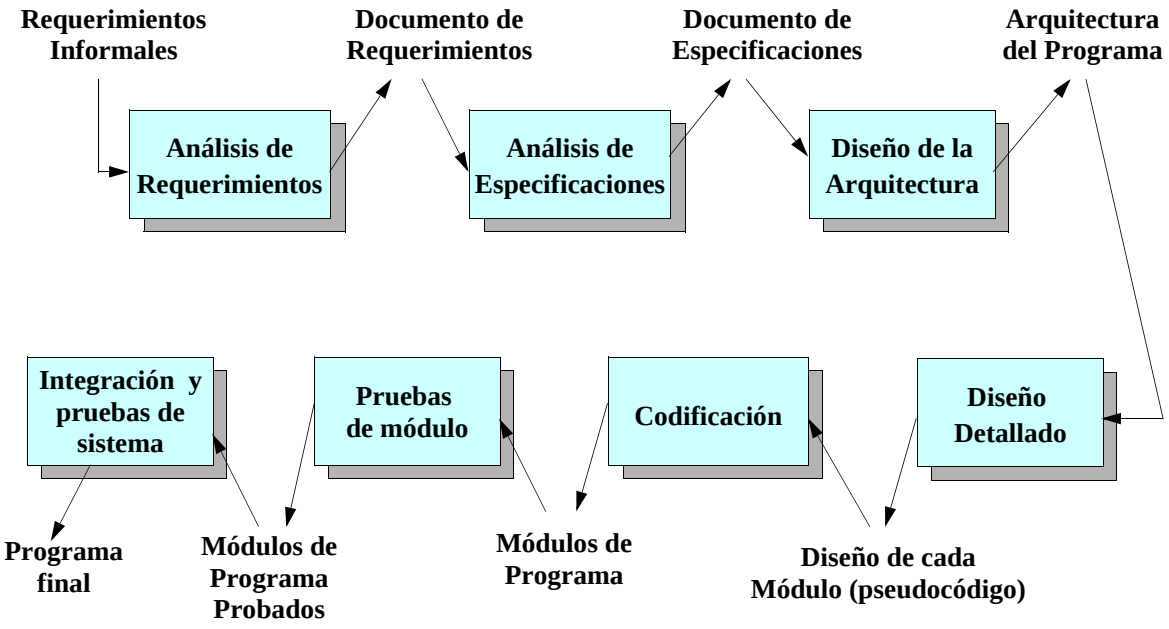
Los programas pueden llegar a ser muy complejos y para poder gestionar esta complejidad se dividen en módulos

- **independientes entre sí**
- **llamados clases en Java**

Un módulo de programa contiene datos y operaciones para manipular datos

Cada una de esas operaciones representa un algoritmo

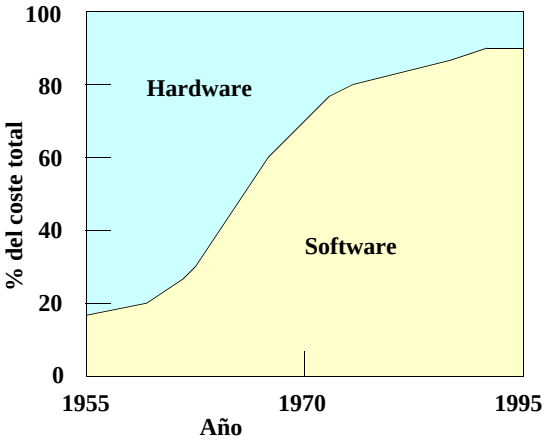
El ciclo de vida de un programa

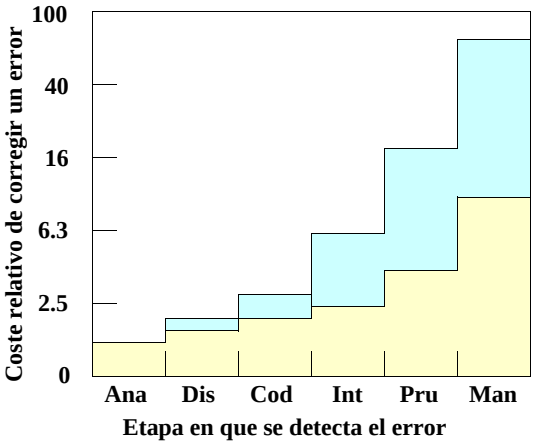


Coste de los sistemas informáticos

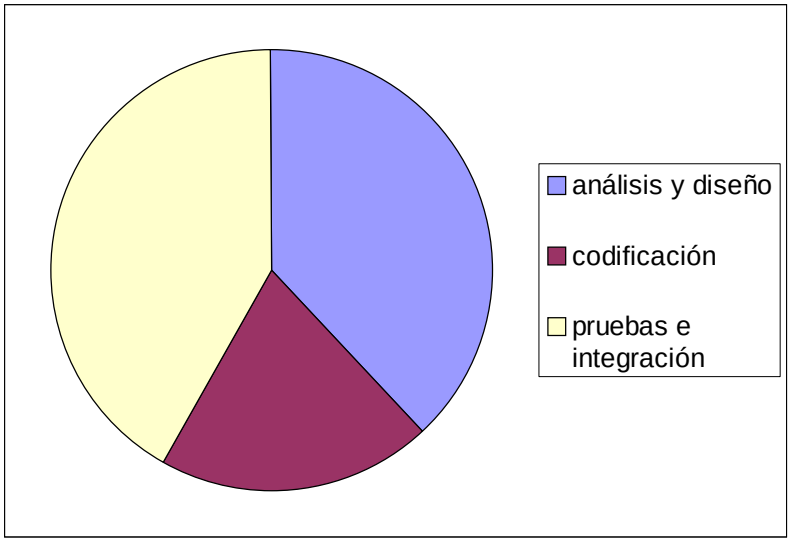
La mayor parte del gasto en sist. informáticos es el software

Los errores software tienen un alto coste: efecto y corrección





Distribución del Esfuerzo de desarrollo



Concepto de algoritmo

Un algoritmo es:

- una secuencia finita de instrucciones,
- cada una de ellas con un claro significado,
- que puede ser realizada con un esfuerzo finito
- y en un tiempo finito

El algoritmo se diseña en la etapa de diseño detallado y se corresponde habitualmente con el nivel de operación o método

Los programas se componen habitualmente de muchas clases que contienen algoritmos, junto con datos utilizados por ellos

- los datos y algoritmos relacionados entre sí se encapsulan en la misma clase

Noción de proceso

Un proceso es la acción de ejecutar las instrucciones de un programa

Se manifiesta por una sucesión de cambios de estado en la memoria y en el entorno del computador

- **es decir, cambios en la información almacenada en el computador y su entorno**

Variables y estado de un programa

La información que maneja un programa se almacena en unidades llamadas variables

- **almacenan un valor**
- **de un determinado tipo**
- **se identifican por un nombre**

El estado de un programa en un momento dado es el valor de las variables en ese instante

Ejemplo del estado de un programa

Programa que intenta mantener una habitación a una temperatura deseada (± 0.5 grados)

- dispone de un radiador que se puede encender y apagar,
- y un termómetro

Variables:

- `tempDeseada`: magnitud real (ºC)
- `tempAmbiente`: magnitud real (ºC)
- `estadoRadiador`: encendido o apagado

Estado del programa

- valor de `tempDeseada`, `tempAmbiente`, y `estadoRadiador` en cada instante

Ejemplo (cont.)

Algoritmo

```
Repetir continuamente lo siguiente
   si hace frío encender el radiador
   si hace calor apagar el radiador
   esperar un rato
frepetir
```

El algoritmo se repite continuamente (hasta que el usuario apague el sistema)

Ahora debemos refinar este algoritmo para expresarlo en términos de las variables del sistema

Ejemplo (cont.)

Algoritmo refinado

```
Repetir continuamente lo siguiente
   si tempAmbiente<tempDeseada-0.5 entonces
      // hace frío
      estadoRadiador:=encendido
   fsi
   si tempAmbiente>tempDeseada+0.5 entonces
      // hace calor
      estadoRadiador:=apagado
   fsi
   esperar 1 minuto
frepetir
```

Observar que si no hace ni frío ni calor el radiador se queda como estaba

A observar

Hemos descrito el algoritmo mediante la técnica llamada pseudocódigo, que tiene

- instrucciones de control presentes en todos los lenguajes

```
si condición entonces
   hacer cosas
fsi
```

- obsérvese el uso del sangrado para determinar el ámbito de aplicación de cada instrucción de control
- cálculos
- acciones expresadas sin el formalismo de los lenguajes

El propósito es que el pseudocódigo refinado sea sencillo, y directamente traducible a código en cualquier lenguaje

Traza del proceso

Podemos observar los diferentes estados del sistema, por ejemplo al inicio del algoritmo

tempDeseada	tempAmbiente	estadoRadiador
23.0	21.4	apagado
23.0	22.1	encendido
23.0	22.8	encendido
23.0	23.4	encendido
23.0	24.1	encendido
23.0	23.8	apagado
24.0	23.3	apagado
24.0	24.2	encendido