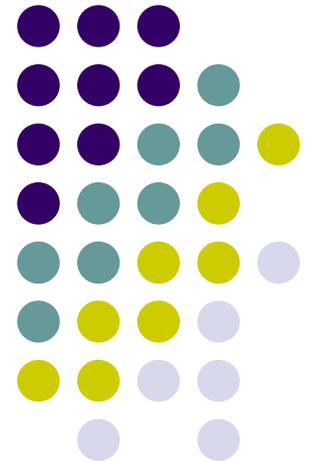


<http://webs.um.es/frgarcia/teaching.htm>

# ALGORITMOS Y ESTRUCTURAS DE DATOS

PRÁCTICA. TEMAS 2 Y 3  
SEMINARIO DE C++  
SESIÓN 2

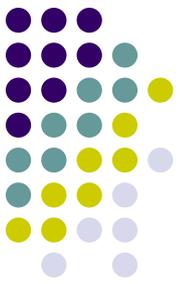


Francisco García Sánchez  
[frgarcia@um.es](mailto:frgarcia@um.es)  
Despacho 2.31

# INDICE DE CONTENIDOS



- Introducción a C++
  - Definición de clases
  - Implementación de los métodos
  - Constructores y destructores
  - Objetos en memoria dinámica
  - `cin` en la práctica
  
- Planificación práctica



# Introducción a C++

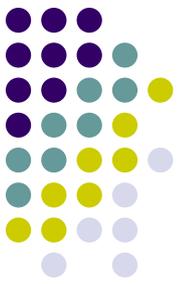
- Definición de clases:
  - **Modelo imperativo**: programa  $\equiv$  sucesión de instrucciones secuenciales o iterativas
  - **Modelo orientado a objetos**: programa  $\equiv$  conjunto de objetos que se comunican entre sí
  - **Objeto**: variable cuyo tipo es una clase

```
class persona // persona es una clase
{
    char nombre[80];
    long dni, telefono;
    int edad;

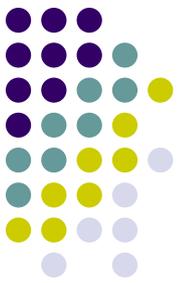
    void leer (FILE *f);
    void escribir (void);
};

persona p1; // p1 es un objeto de clase persona
p1.leer(f1);
p1.edad++;
p1.escribir();
```

# Introducción a C++



- Definición de clases:
  - Nomenclatura:
    - **Miembros de una clase:** atributos y operaciones de la clase (datos y funciones); p.ej. `dni` y `edad`
    - **Métodos de una clase:** funciones definidas dentro de la clase; p.ej. `leer()` y `escribir()`
    - **Mensaje:** invocación de un método sobre un objeto; p.ej. `p1.leer(f1)`
    - **Objeto receptor:** objeto sobre el que se aplica un mensaje; p.ej. en `p1.leer(f1)` el receptor es `p1`.



# Introducción a C++

- Definición de clases:

```
class nombre {  
    ...  
};
```

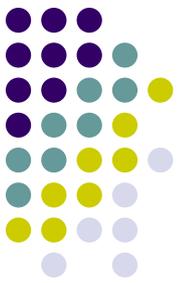
- **Declaración:**

- **Miembros públicos y privados:**

```
class nombre {  
    private:  
        ...           // Miembros privados  
    public:  
        ...           // Miembros públicos  
};
```

- **Declaración de los miembros:**

```
class conjuntoInt {  
    private:  
        int tamano;  
        int datos[MAXIMO];  
    public:  
        void vaciar ();  
        void insertar (int n);  
        void suprimir (int n);  
        bool miembro (int n);  
};
```



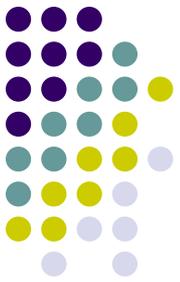
# Introducción a C++

- Implementación de los métodos:
  - Generalmente fuera de la definición de la clase
    - Operador de visibilidad: ‘ :: ’

```
void conjuntoInt::vaciar ()  
{  
    ...  
}
```

```
void conjuntoInt::insertar (int n)  
{  
    ...  
}
```

- Definiendo el método **todos** los miembros de la clase son accesibles (como variables locales).
- **Métodos in-line:**
  - Método dentro de la definición de la clase.
  - No aconsejable, a menos que sea trivial.



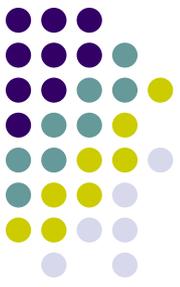
# Introducción a C++

- Implementación de los métodos:
  - Clase conjunto de enteros con tamaño máximo limitado

*Ejemplo página 5*

*Cambiar #define  
por const int*





# Introducción a C++

- Constructores y destructores:
  - **Constructor**: operación de inicialización de un objeto.
    - Por defecto, sin constructor, datos no se inicializan.
    - **Constructor**  $\equiv$  método con mismo nombre que clase
    - **Sobrecarga** permitida:

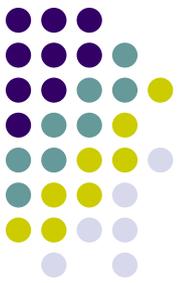
```
class conjuntoInt {  
    ...  
public:  
    conjuntoInt () {tamano= 0;} // Constructor de conjunto vacio  
    conjuntoInt (int e1) // Constructor de cjt. con 1 elem.  
        {datos[0]= e1; tamano= 1;}
```

**constructor por defecto**



- **Uso constructores:**

```
conjuntoInt cjt; // Se inicializa con el constructor por defecto  
conjuntoInt cjt(9); // Constructor que incluye 1 elemento
```

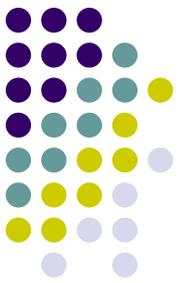


# Introducción a C++

- Constructores y destructores:
  - **Destructor**: operación de eliminación de un objeto
    - Nombre del destructor: “**~NombreClase**”
    - No recibe parámetros; no devuelve ningún valor
    - Necesario si el objeto ha reservado memoria dinámica o abre algún fichero.
      - **Programador responsable de liberar/cerrar**

```
class conjuntoInt {  
    ...  
    public:  
        ~conjuntoInt () {cout<<"Nada";} // Destructor, irrelevante aquí  
    ...  
};
```

```
cjt.~conjuntoInt();
```



# Introducción a C++

- Objetos en memoria dinámica:
  - Igual a otros tipos de datos
    - Operadores: `new`, `new []`, `delete` y `delete []`
    - Crea objetos en memoria dinámica...necesario liberar memoria cuando objeto ya no usado.

```
nombreClase *obj1= new nombreClase;
```

→ Crear un nuevo objeto usando el constructor por defecto.

```
nombreClase *obj2= new nombreClase(p1, p2, ...);
```

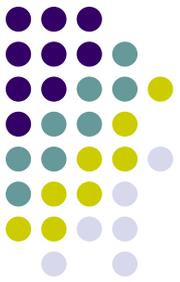
→ Crear un nuevo objeto usando un constructor específico.

```
nombreClase *ad= new nombreClase[tamano];
```

→ Crear un array dinámico de objetos con el constructor por defecto.

```
delete obj1; → Borra el objeto dinámico, usando el destructor.
```

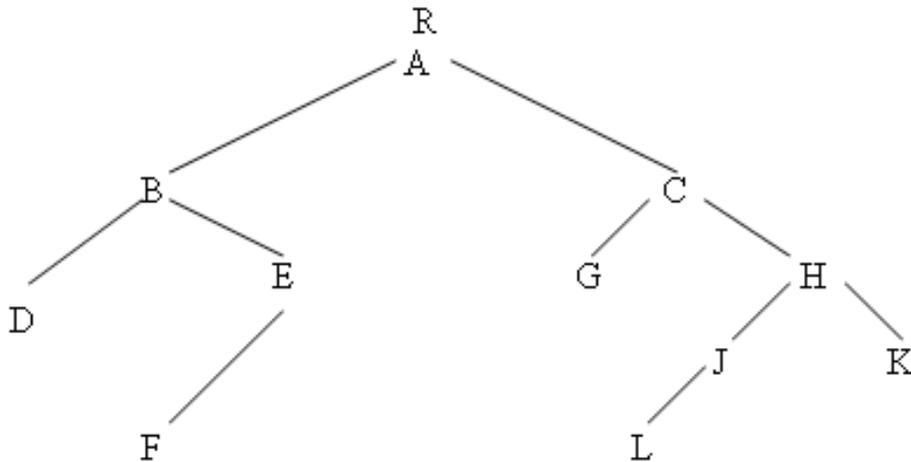
```
delete[] ad; → Borra un array dinámico de objetos, usando el destructor.
```

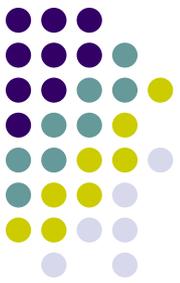


# Introducción a C++

- Objetos en memoria dinámica:
  - Árbol binario de cadenas

*Ejemplo página 8*





# Introducción a C++

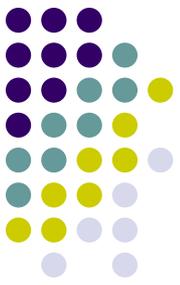
## ● **cin:**

- Objeto de la clase 'istream'
  - <http://www.cplusplus.com/reference/iostream/istream/>
  - `cin.eof()` – marca fin de fichero
  - `cin.get()` – obtener datos sin formato
- Obtener cadena en una línea:

- Librería `<string>`
- `getline(istream& is, string& str)`

```
// getline with strings
#include <iostream>
#include <string>
using namespace std;

int main () {
    string str;
    cout << "Please enter full name: ";
    getline (cin, str);
    cout << "Thank you, " << str << ".\n";
}
```



# Planificación práctica

- Abierto concurso: “**AED1 12/13. Práctica**”
  - Cierra: 13 de diciembre a las 14:00 horas.
  - Total 12 ejercicios obligatorios.
  - 2 ejercicio opcionales (por incluir en concurso)
  - Mismas claves; 1 persona por grupo



# Planificación práctica

**Semana 2.** 29 de octubre: introducción a C++ y ejercicios previos: separar palabras (001), palabras desde una cadena (002), convertir a minúsculas (002) y la clase producto (004).

- 001
  - Dado un texto, separar las **palabras** que contiene
  - **Palabra** = cualquier sucesión de letras, números o símbolos separados por espacios en blanco
  - ❖ *Los espacios en blanco son el carácter de espacio " ", el tabulador "\t" y el fin de línea "\n".*

Entrada

Pascuard Bell Viseo230 23" Full HD



Salida

Pascuard  
Bell  
Viseo230  
23"



# Planificación práctica

**Semana 2.** 29 de octubre: introducción a C++ y ejercicios previos: separar palabras (001), palabras desde una cadena (002), convertir a minúsculas (002) y la clase producto (004).

- 002
  - Separación de palabras sobre **cadena de texto**
  - Recoger líneas con **getline**
  - Procesar líneas con **istringstream**

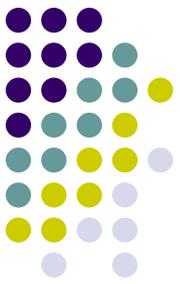
Entrada

```
insertar 84021349837
Pascuard Bell Viseo230 23"
```



Salida

```
1. insertar 84021349837
insertar
84021349837
2. Pascuard Bell Viseo230 23"
Pascuard
Bell
```



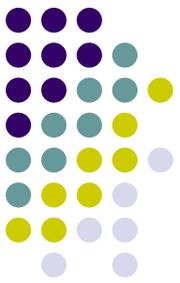
# Planificación práctica

- **getline:**

- <http://www.cplusplus.com/reference/string/getline/>
- Librería <string>
- `getline(istream& is, string& str)`

```
// getline with strings
#include <iostream>
#include <string>
using namespace std;

int main () {
    string str;
    cout << "Please enter full name: ";
    getline (cin, str);
    cout << "Thank you, " << str << ".\n";
}
```



# Planificación práctica

## ● **istringstream:**

- <http://www.cplusplus.com/reference/iostream/istringstream/>
- <http://www.cplusplus.com/reference/iostream/istringstream/istringstream/>

```
// using istringstream constructors.
#include <iostream>
#include <sstream>
#include <string>
using namespace std;

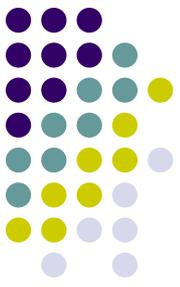
int main () {

    int n, val;
    string stringvalues;

    stringvalues = "125 320 512 750 333";
    istringstream iss (stringvalues, istringstream::in);

    for (n=0; n<5; n++)
    {
        iss >> val;
        cout << val*2 << endl;
    }

    return 0;
}
```



# Planificación práctica

**Semana 2.** 29 de octubre: introducción a C++ y ejercicios previos: separar palabras (001), palabras desde una cadena (002), convertir a minúsculas (002) y la clase producto (004).

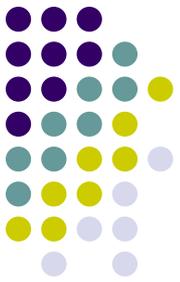
- 003
  - Función para convertir cadena (string) a minúsculas
  - Letras estándar: **tolower** (<[cctype.h](http://www.cplusplus.com/reference/clibrary/cctype/tolower/)>)
    - <http://www.cplusplus.com/reference/clibrary/cctype/tolower/>
  - Letras no estándar (codificación UTF-8): uso de 2 bytes para representación: **0xC3 0x??**

# Planificación

- Normalización

[http://es.wikipedia.org/wiki/Codificaci%C3%B3n\\_de\\_caracteres](http://es.wikipedia.org/wiki/Codificaci%C3%B3n_de_caracteres)

Acentos y tildes					
carácter	ISO-8859-1	UTF-8		UTF-16	
á	0xe1	0xc3	0xa1	0x00	0xe1
Á	0xc1	0xc3	0x81	0x00	0xc1
é	0xe9	0xc3	0xa9	0x00	0xe9
É	0xc9	0xc3	0x89	0x00	0xc9
í	0xed	0xc3	0xad	0x00	0xed
Í	0xcd	0xc3	0x8d	0x00	0xcd
ó	0xf3	0xc3	0xb3	0x00	0xf3
Ó	0xd3	0xc3	0x93	0x00	0xd3
ú	0xfa	0xc3	0xba	0x00	0xfa
Ú	0xda	0xc3	0x9a	0x00	0xda
ü	0xfc	0xc3	0xbc	0x00	0xfc
Ü	0xdc	0xc3	0x9c	0x00	0xdc
ñ	0xf1	0xc3	0xb1	0x00	0xf1
Ñ	0xd1	0xc3	0x91	0x00	0xd1



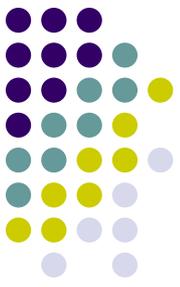
# Planificación práctica

## Ejemplo de Entrada

```
insertar 84021349837
Pascuard BELL Viseo230 23" Full HD
Monitor Full HD de 23" con un precio INCREÍBLE . Alta definición . Acabado elegante .
99.95
palabras monitor bell PEQUEÑA DEFINICIÓN
precios 80 99.99
```

## Ejemplo de Salida

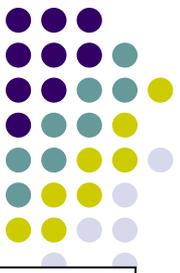
```
1. insertar 84021349837
insertar
84021349837
2. Pascuard BELL Viseo230 23" Full HD
pascuard
bell
viseo230
23"
full
hd
3. Monitor Full HD de 23" con un precio INCREÍBLE . Alta definición . Acabado elegante .
monitor
full
hd
de
23"
con
un
precio
increíble
.
alta
```



# Planificación práctica

**Semana 2.** 29 de octubre: introducción a C++ y ejercicios previos: separar palabras (001), palabras desde una cadena (002), convertir a minúsculas (002) y la clase producto (004).

- 004
  - TAD Producto
    - **Datos**: nombre, identificador, descripción, precio
    - **Funciones**: leer, almacenar, consultar, escribir, ...
  - Entrada: número arbitrario de comandos “**insertar**”



# Planificación práctica

## Ejemplo de Entrada

```
insertar 84021349837
Pasuard BELL Viseo230 23" Full HD
Monitor Full HD de 23" con un precio INCREÍBLE . Alta definición . Acabado elegante .
99.95
insertar 8403787362543
G-Data Internet Security 2012 3 Licencias 2 Años
Protección antivirus vencedora en numerosas pruebas y análisis independientes.
14.95
insertar 8982377636748
SanDisk MicroSDXC 64GB Mobile Ultra Clase 10 Android Edition
Disfrute de más velocidad y una capacidad máxima de almacenamiento para fotos
44.0
```

## Ejemplo de Salida

1. Pasuard BELL Viseo230 23" Full HD (84021349837)  
Monitor Full HD de 23" con un precio INCREÍBLE . Alta definición . Acabado elegante .  
99.95 euros
2. G-Data Internet Security 2012 3 Licencias 2 Años (8403787362543)  
Protección antivirus vencedora en numerosas pruebas y análisis independientes.  
14.95 euros
3. SanDisk MicroSDXC 64GB Mobile Ultra Clase 10 Android Edition (8982377636748)  
Disfrute de más velocidad y una capacidad máxima de almacenamiento para fotos  
44 euros