

# TECNOLOGIAS DE INFORMACIÓN PARA LA INNOVACIÓN

---

# Bibliografía

Autor(es) Apellido(S),Nombre(S)	Título del libro	Editorial	Año
Bhambri P. Y Singh S.	Fundamentals Of Information Technology: Introduction To Applications Of It	Lap Lambert Academic Publishing, 2013 [L] [SEP]	2013
Ramajaran, V.	Introduction To Information Technology	Prentice-Hall Of India, 2004	2004
Williams, Brian.	Using Information Technology.	Career Education; 8th Ed., 2009	2009

# Bibliografía. Clase

---

Sharma DHIRAJ, Foundations of IT. Excel Books, 2008. ISBN 978-81-7446-589-4

Turban Efraim, McClean Ephraim et al. Tecnologías de Información para la Administración. CECSA 2006 ISBN 970-24-0025-2

Elmasri Ramez, Navathe Shamkant. Fundamentos de Sistemas de Bases de Datos. Addison-Wesley. 2005.

Tecnologías de la información y las comunicaciones. Consultado el 25 de junio de 2017, desde: [https://www.ecured.cu/Tecnolog%C3%ADas de la informaci%C3%B3n y las comunicaciones](https://www.ecured.cu/Tecnolog%C3%ADas_de_la_informaci%C3%B3n_y_las_comunicaciones)

Sarukkai Ramesh. Foundations of Web Technology. Springer 2002. ISBN 978-1-4615-5409-3

Alan Dix, Janet Finlay, Gregory Abowd & Russell Beale. Human-Computer Interaction. 3rd Edition. Prentice Hall, 2004. ISBN 0-13-046109-1.

Web technology for developers. Consultado el 25 de junio de 2017, desde: <https://developer.mozilla.org/en-US/docs/Web>

# Agenda

---

## Unidad II. Tecnología de Información

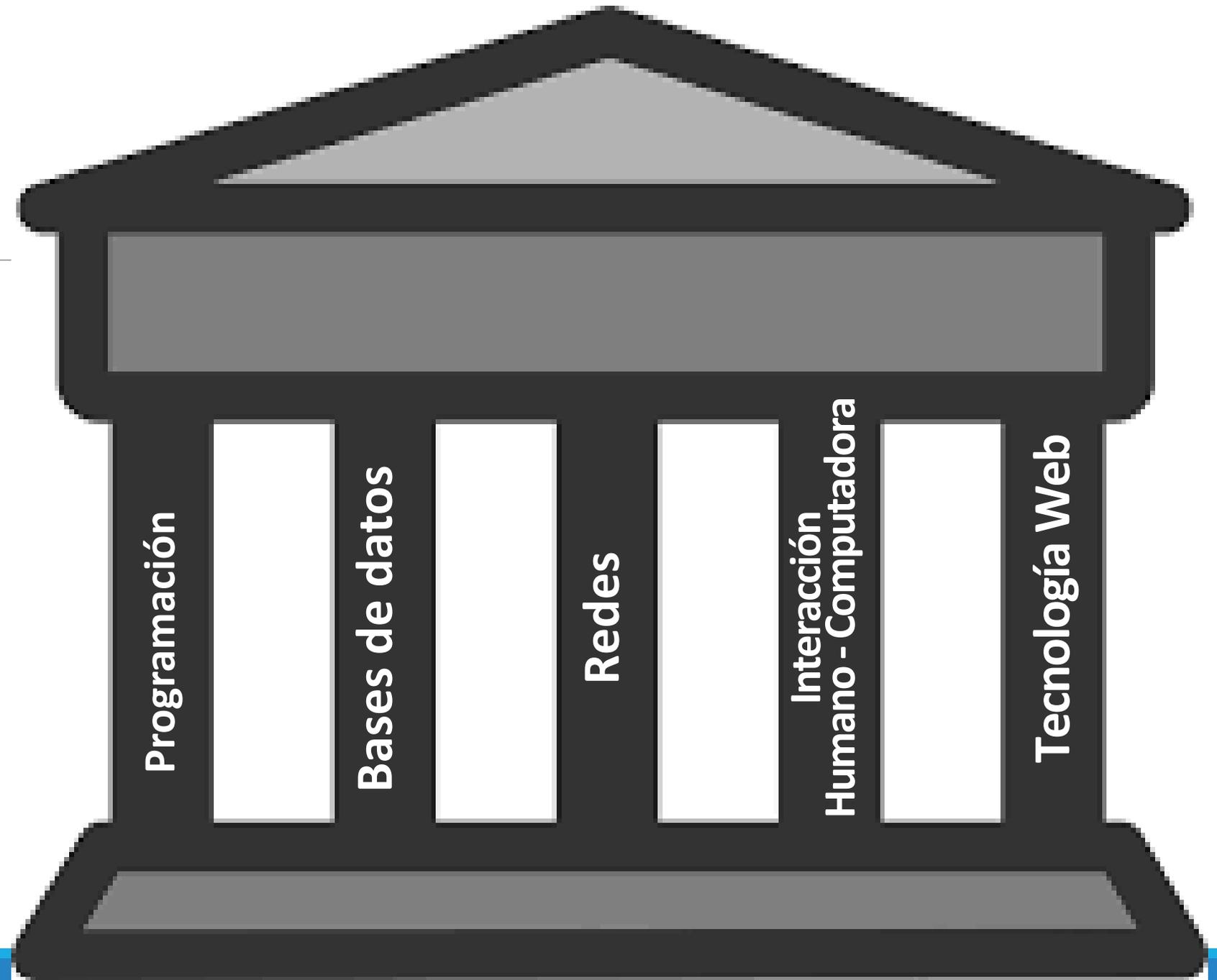
- I. Definición
- II. Pilares: **programación**, bases de datos, redes, interacción humano-computadora, tecnologías web

# Tecnologías de información, Definición

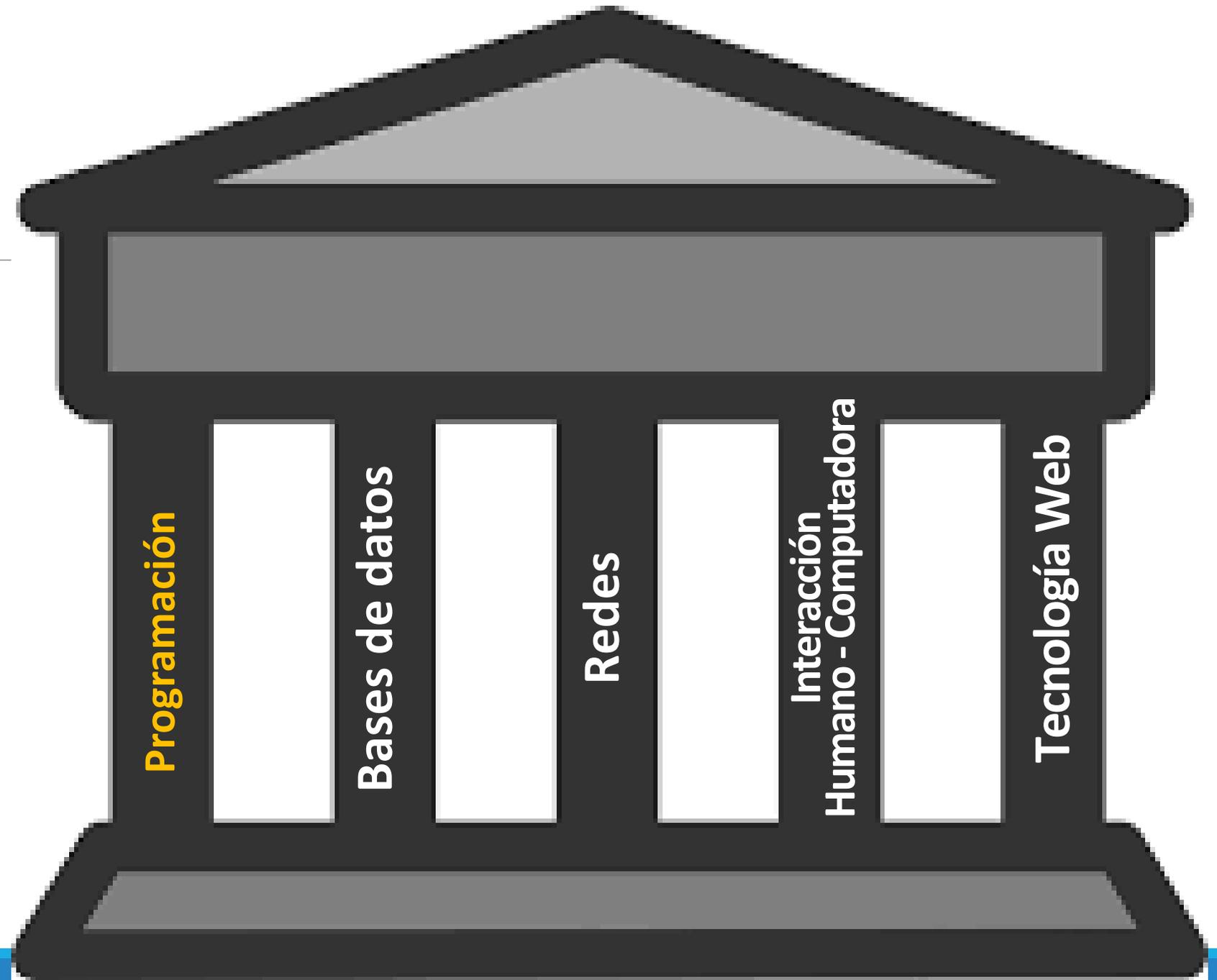
---

- ❑ “IT o Tecnología de la información concierne a herramientas, técnicas y tecnología aplicada para la transmisión, almacenamiento, procesamiento y diseminación de información.” Sharma (2008)
- ❑ La Information Technology Association of America (ITAA) define Tecnología de Información como “El estudio, diseño, desarrollo, implementación, soporte o administración de sistemas de información basados en computadora, particularmente aplicaciones de software y hardware de computadora”

# Pilares de la TI



# Pilares de la TI



# Pilares de la TI. Programación.

---

## ■ Programación.

➤ Es el proceso a través del cual las instrucciones para la computadora son ordenadas y agrupadas en forma lógica para construir un programa; con el fin de mandar a ejecutar ciertas operaciones.



# Pilares de la TI. Programación.

---

¿Cómo entiende la computadora un programa elaborado en lenguaje de alto nivel?

# Pilares de la TI. Programación.

---

Un programa elaborado en lenguaje de alto nivel, no puede ser ejecutado directamente por una computadora, sino que debe ser traducido a lenguaje máquina.

# Pilares de la TI. Programación.

---

- **Programa fuente:** Programa escrito en un lenguaje de alto nivel y necesita ser traducido a código máquina para ser ejecutado.
- **Compilador:** Programa encargado de traducir los programas fuentes escritos en un lenguaje de alto nivel a código objeto. La traducción del programa completo se realiza en una sola operación denominada compilación del programa
- **Intérprete:** Es un compilador que traduce una instrucción de programa fuente a la vez.

# Pilares de la TI. Programación.

---

- **Programa objeto:** Es el programa fuente traducido (por el compilador) a código máquina. Aún no es directamente ejecutable.
- **Programa Ejecutable:** Traducción completa a código máquina, realizada por el enlazador, del programa fuente y que ya es directamente ejecutable.
- **Enlazador:** programa encargado de insertar al programa objeto el código máquina de las funciones de las librerías (archivos de biblioteca) usadas en el programa y realizar el proceso de montaje, que producirá un programa ejecutable .exe

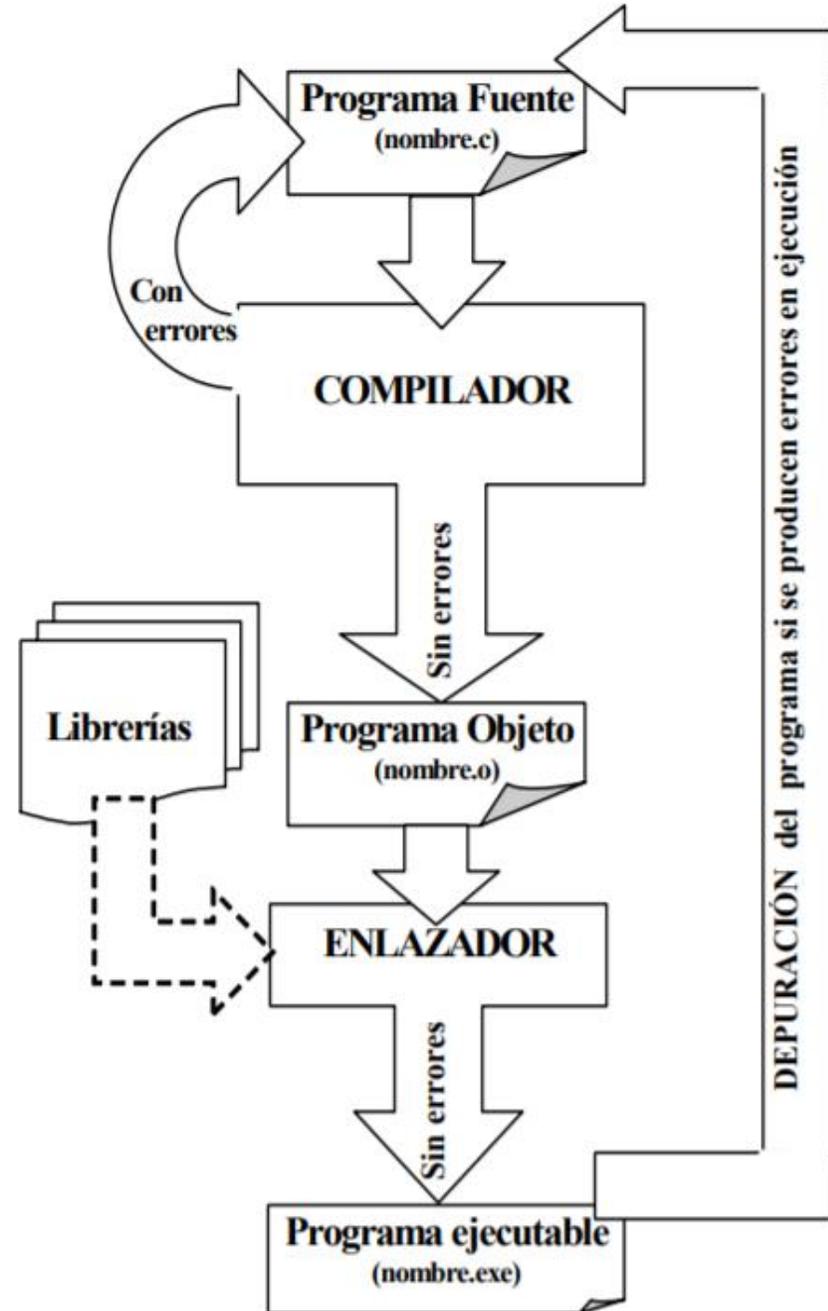
# Pilares de la TI. Programación.

---

- Las **librerías** son una colección de código (funciones) ya programado y traducido a código máquina, listo para utilizar en un programa y que facilita la labor del programador.

# Pilares de la TI. Programación

- Proceso de transformación de un programa fuente a un programa ejecutable



# Pilares de la TI. Programación.

---

## Ejemplo:

Código de operación	Dirección	Instrucción en lenguaje ensamblador	Instrucción en lenguaje de alto nivel
00010101	10000001	LOAD A	
00010111	10000010	ADD B	
00010110	10000011	STORE C	C = A + B

# Pilares de la TI. Programación.

hola.c

```
#include <stdio.h>

int main()
{
    printf("Hola Mundo!\n");
    return 0;
}
```

Compilador  
C

Programa objeto  
Hola.obj

```
.model small
.stack
.data
Cadena1 DB 'Hola Mundo.$'
.code

programa:
    mov ax, @data
    mov ds, ax
    mov dx, offset Cadena1
    mov ah, 9
    int 21h
end programa
```

Enlazador

hola.exe

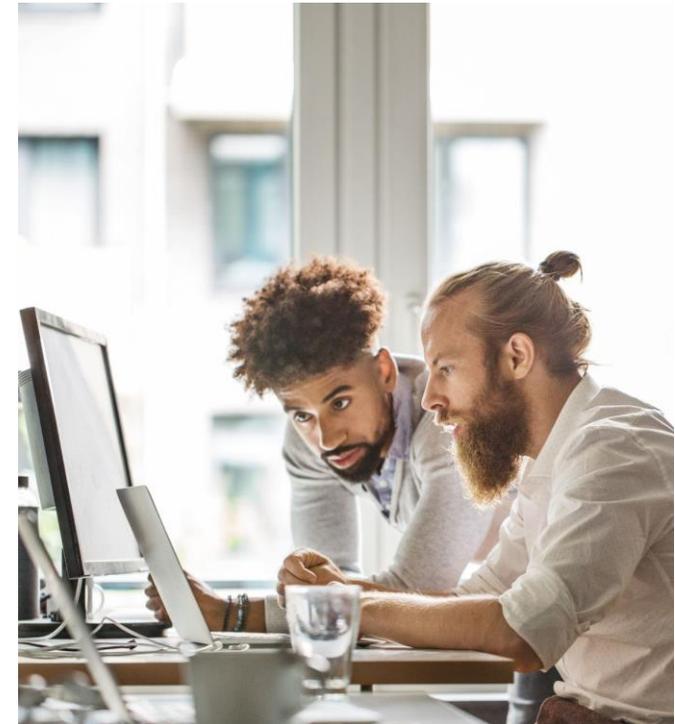
```
100101101010
001001011110
101001000101
010100101010
101010010101
010101010101
000000110010
```

# Pilares de la TI. Programación.

---

## ■ Programación.

➤ Es el proceso a través del cual las instrucciones para la computadora son ordenadas y agrupadas en forma lógica para construir un programa; con el fin de mandar a ejecutar ciertas operaciones.



# Paradigmas de la programación

---

# Pilares de la TI. Programación.

---

- Programación Estructurada
- Programación Modular
- Programación Orientada a Objetos.

# Pilares de la TI. Programación.

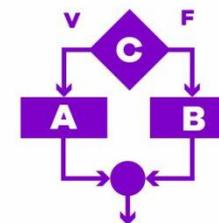
## ■ Programación Estructurada.

- C, Pascal, FORTRAN, y lenguajes similares, se conocen como lenguajes procedimentales (por procedimientos).
- Cada sentencia o instrucción señala al compilador para que realice alguna tarea: obtener una entrada, producir una salida, sumar tres números, dividir por cinco, etc.
- En el caso de pequeños programas esto es eficiente.

Secuencia



Selección o condicional



Iteración (ciclo o bucle)



# Pilares de la TI. Programación.

---

## ■ Programación Estructurada.

- Cuando los programas se vuelven más grandes, la lista de instrucciones aumenta considerablemente, de modo tal que el programador tiene muchas dificultades para controlar ese gran número de instrucciones.
- Los programadores pueden controlar, de modo normal, unos centenares de líneas de instrucciones.

# Pilares de la TI. Programación.

---

## ■ Programación Estructurada.

- Para resolver este problema los programas se descompusieron en unidades más pequeñas que adoptaron el nombre de funciones
- Un programa orientado a procedimientos se divide en funciones.
- Cada función tiene un propósito bien definido y resuelve una tarea concreta.
- Con el paso de los años, la idea de romper el programa en funciones fue evolucionando y se llegó al agrupamiento de las funciones en otras unidades más grandes llamadas **módulos**.

```
1 //Uso del paso por referencia con un argumento apuntador para elevar al cubo una variable
2 #include <iostream>
3 using namespace std;
4
5 void cuboPorReferencia( int * ); // prototipo
6
7 int main()
8 {
9     int numero = 5;
10
11     cout << "El valor original de numero es " << numero;
12
13     cuboPorReferencia( &numero ); // pasa la dirección de numero a cuboPorReferencia
14
15     cout << "\nEl nuevo valor de numero es " << numero << endl;
16     return 0; // indica que terminó correctamente
17 } // fin de main
18
19 // calcula el cubo de *nPtr; modifica la variable numero en main
20 void cuboPorReferencia( int *nPtr )
21 {
22     *nPtr = *nPtr * *nPtr * *nPtr; // eleva *nPtr al cubo
23 } // fin de la función cuboPorReferencia
```

# Pilares de la TI. Programación.

---

## ■ Programación Modular.

- El programa se divide en módulos (partes independientes), cada uno de los cuales ejecuta una única actividad o tarea y se codifican independientemente de otros módulos.
- Cada uno de estos módulos se analiza, codifica y pone a punto por separado.
- Los módulos son independientes.
- Los resultados producidos por un módulo pueden ser utilizados por cualquier otro módulo.

# Pilares de la TI. Programación.

---

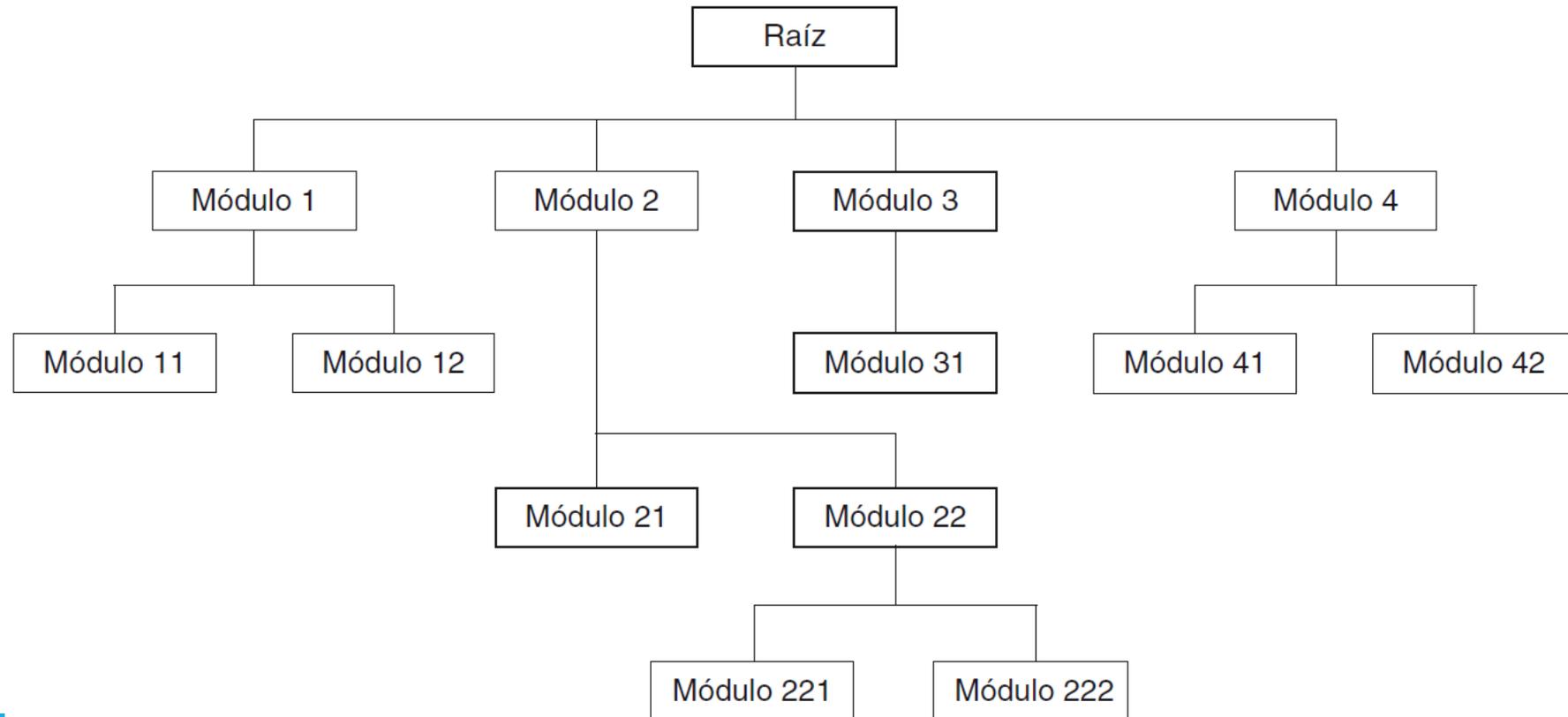
## ■ Programación Modular.

- Cada programa contiene un módulo denominado **programa principal** que controla todo lo que sucede; se transfiere el control a submódulos. de modo que ellos puedan ejecutar sus funciones; sin embargo, cada submódulo devuelve el control al módulo principal cuando se haya completado su tarea.

# Pilares de la TI. Programación.

---

- Programación Modular.



■ Módulos y sub módulos

□ Funcionalidad

■ Comportamiento

# Sistema de Cursos en línea

## Usuarios

Registro, Modificación, Eliminación, Asignación de cursos...

## Actividades

Registro, Modificación, Eliminación, Calificación, Asignación de cursos

## Exámenes

Creación, Modificación, Eliminación, Asignación, Calificación

## Registro

Registro de usuarios

## Modificación

Modificación de usuarios

## Eliminación

Eliminación de usuarios

## Asignación

Asignación de usuarios a grupos

# Pilares de la TI. Programación.

---

## ■ Programación Modular.

- Diferentes programadores pueden trabajar simultáneamente en diferentes partes del mismo programa.
- Un módulo se puede modificar radicalmente sin afectar a otros módulos.
- La descomposición de un programa en módulos independientes más simples se conoce también como el método de **divide y vencerás**.

# Pilares de la TI. Programación.

---

## ■ Programación Orientada a Objetos.

- El paradigma de programación más utilizado en el mundo del desarrollo de software y de la ingeniería de software del siglo XXI.
- Al contrario que la programación procedimental que enfatiza en los algoritmos, la POO enfatiza en los datos.
- En lugar de intentar ajustar un problema al enfoque procedimental de un lenguaje, POO intenta ajustar el lenguaje al problema.
- La idea es diseñar formatos de datos que se correspondan con las características esenciales de un problema.

# Pilares de la TI. Programación.

---

## ■ Programación Orientada a Objetos.

- La idea fundamental de los lenguajes orientados a objetos es combinar en una única unidad o módulo, tanto los datos como las funciones que operan sobre esos datos.
- Tal unidad se llama un objeto.
- Las funciones de un objeto se llaman Métodos. Y son el único medio para acceder a sus datos.
- Los datos de un objeto, se conocen también como atributos.

# Pilares de la TI. Programación.

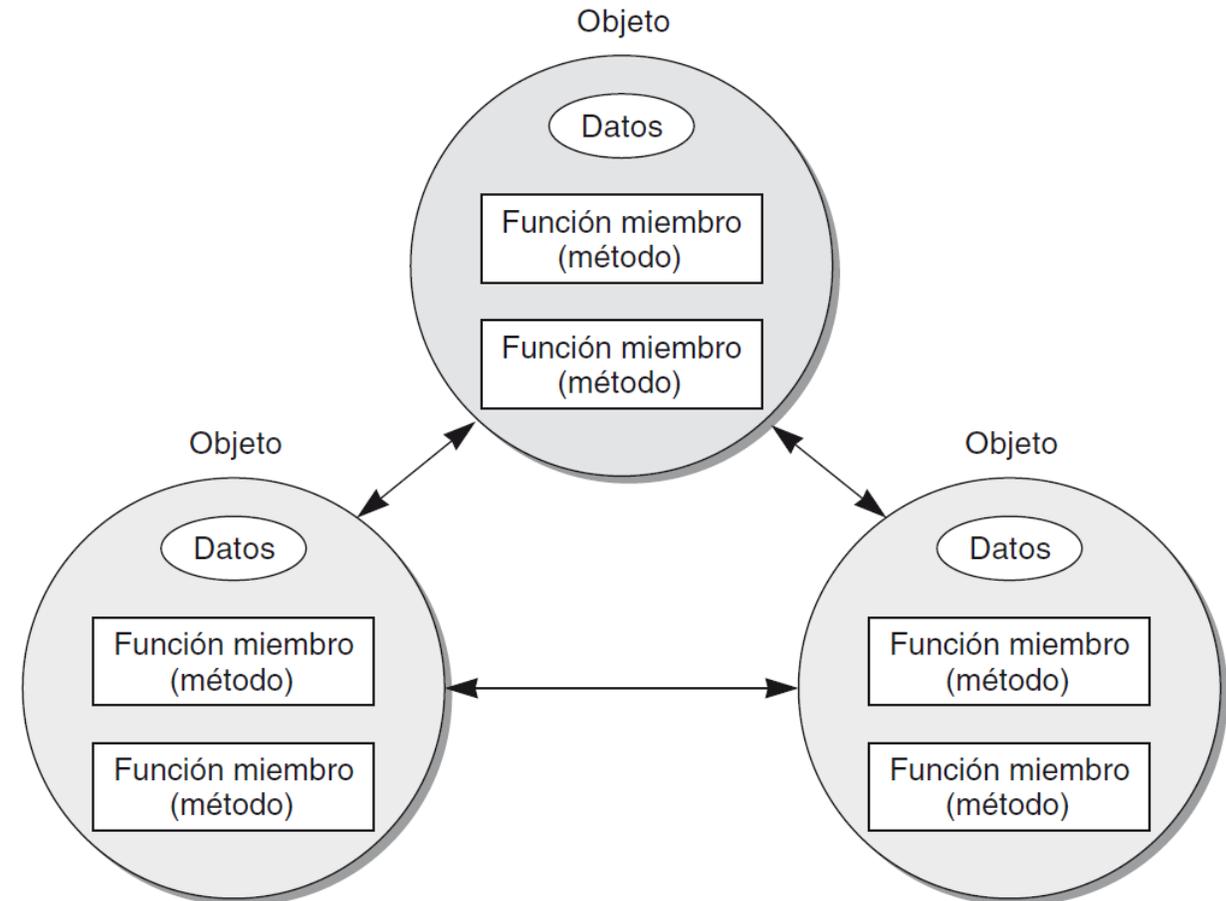
---

## ■ Programación Orientada a Objetos.

- Si se desea leer datos de un objeto, se llama al método del objeto. Se accede a los datos y se devuelve un valor.
- No se puede acceder a los datos directamente. Los datos están ocultos, de modo que están protegidos de alteraciones accidentales.
- Los datos y las funciones se dice que están encapsulados en una única entidad.
- El encapsulamiento de datos y la ocultación de los datos son términos clave en la descripción de lenguajes orientados a objetos.

# Pilares de la TI. Programación.

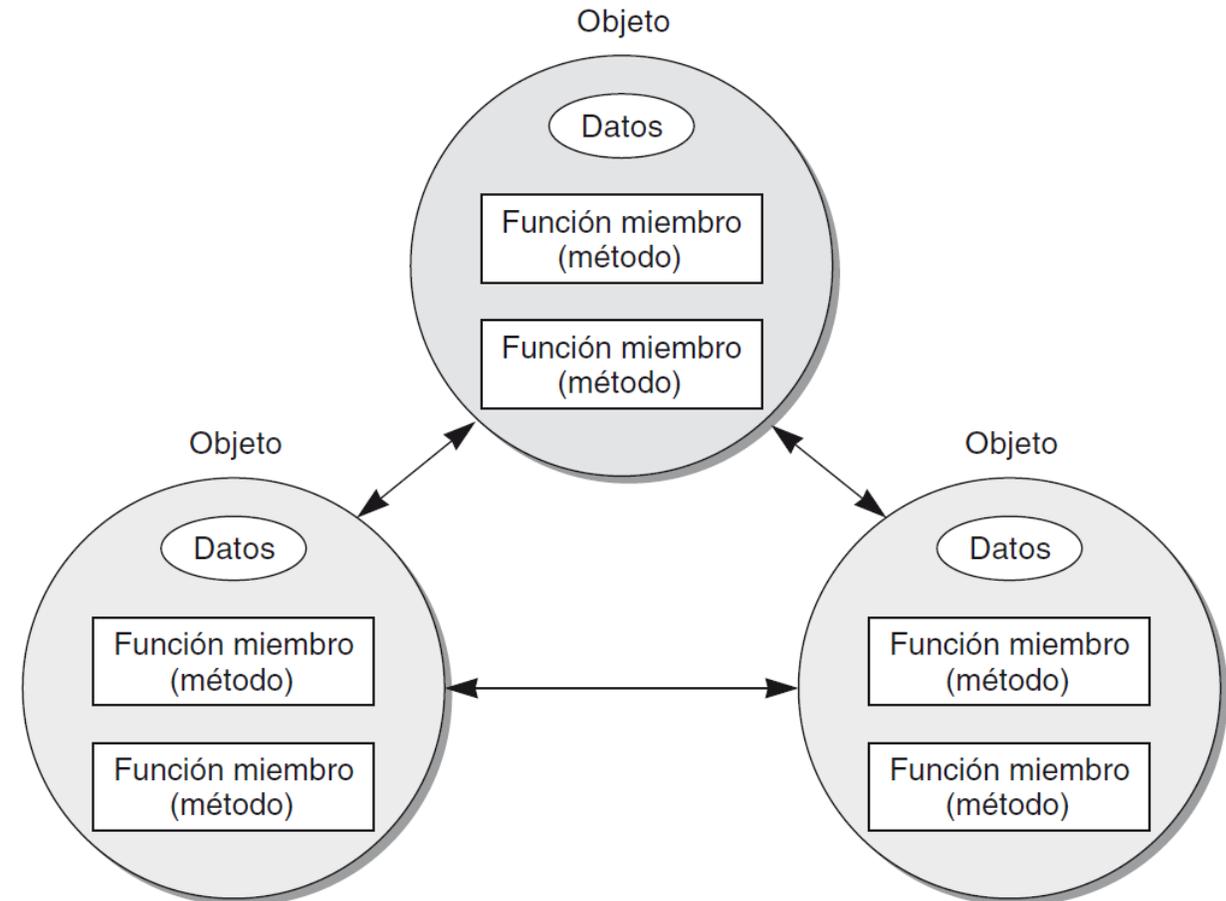
En el paradigma orientado a objetos, el programa se organiza como un conjunto finito de objetos que contiene datos y operaciones (*métodos*) que llaman a esos datos y que se comunican entre sí mediante mensajes.



# Pilares de la TI. Programación.

Propiedades de la OO:

- Abstracción (tipos abstractos de datos y clases).
- Encapsulado de datos.
- Ocultación de datos.
- Herencia.
- Polimorfismo.



# Pilares de la TI. Programación.

---

## Introducción

### Objeto

Los objetos son representaciones de cosas: reloj, avión, coche.

No todo puede ser considerado como un objeto, algunas cosas son simplemente características o atributos de los objetos: color, velocidad, nombre.



# Pilares de la TI. Programación.

---

## Introducción

## Objeto

## Abstracción funcional

Hay cosas que sabemos que los coches hacen pero no cómo lo hacen:

- Avanzar
- Parar
- Girar a la derecha

# Pilares de la TI. Programación.

---

## Introducción

## Objeto

### Abstracción de datos

Un coche tiene además ciertos atributos:

- Color
- Velocidad
- Tamaño

# Pilares de la TI. Programación.

---

## Introducción

### Clase

Conjunto de objetos con estados y comportamientos similares.

“Mi coche” es un objeto, es decir una instancia particular de la clase coche.

Un objeto se distingue de otros miembros de la clase por sus atributos

# Pilares de la TI. Programación.

---

## Introducción

Las cosas que nos rodean...

- ❖ Tienen atributos
- ❖ Realizan acciones

También se agrupan en categorías (automóviles, mobiliario, lavadoras).

Estas categorías son las **clases**.

# Pilares de la TI. Programación.

---

## Introducción

Una **clase** es una categoría o grupo de cosas que tienen atributos y acciones similares.

# Pilares de la TI. Programación.

---

## Por ejemplo:

La clase Lavadora. Cualquier cosa dentro de la clase lavadora tendrá marca, modelo, número de serie, capacidad.

Entre las **acciones** que realizan las cosas de esta clase se encuentran: “agregar detergente”, “activarse”, “lavar”, “centrifugar”

Lavadora de Juan



Lavadora de Pedro



Lavadora de Marta



# Pilares de la TI. Programación.

---

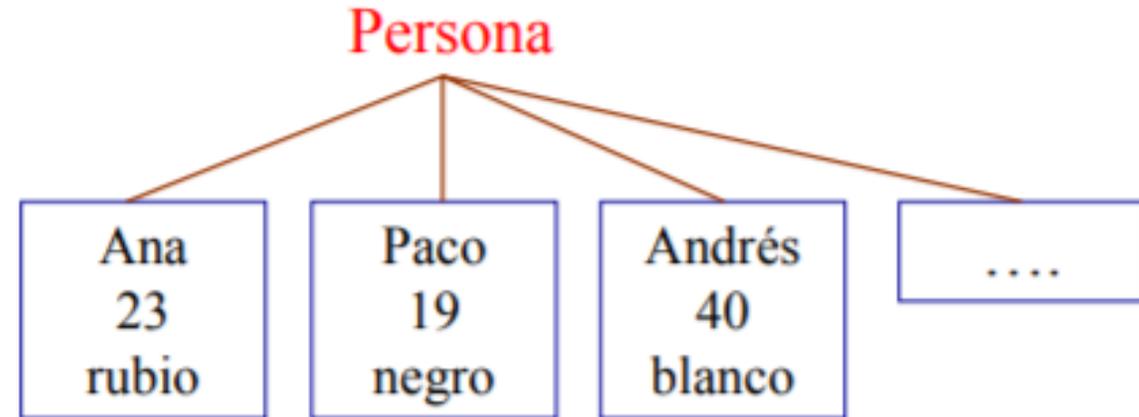
## La clase “Persona”

### Atributos

- Nombre
- Edad
- Color del pelo: (negro, blanco, pelirrojo, rubio)

### Métodos

- Nacer
- Crecer
- Morir



# Pilares de la TI. Programación.

---

**Nombre**

Lavadora

**Atributos**

marca  
modelo  
numero de serie  
capacidad

**Acciones**

agregar ropa()  
agregar detergente()  
sacar ropa()

Representación.

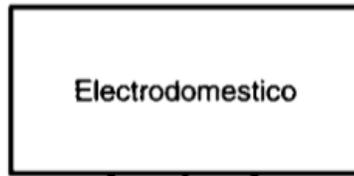
## Lavadora

marca  
modelo  
numeroSerie  
capacidad  
volumenTambor  
cronometroInterno  
trampa  
motor  
velocidadMotor

agregarRopa()  
agregarDetergente()  
sacarRopa()  
agregarBlanqueador()  
cronometrarRemojo()  
cronometrarLavado()  
cronometrarEnjuague()  
cronometrarCentrifugado()

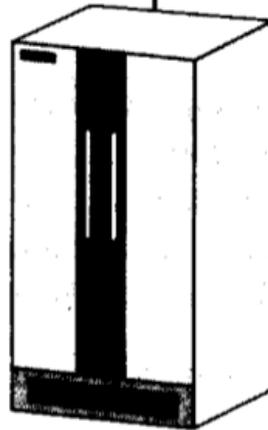
# Herencia

## Superclase



Consiste en factorizar los elementos comunes (atributos, operaciones y restricciones) de un conjunto de clases en una clase más general llamada superclase

## Subclase



Permite capturar particularidades de un conjunto de objetos no discriminados por las clases ya identificadas. Las nuevas características se representan por una nueva clase, subclase de una de las clases existentes.

Los atributos, operaciones y asociaciones de la clase padre están disponibles en sus clases hijas.

# Pilares de la TI. Programación.

---

La clase **Estudiante**.



Estudiante

# Pilares de la TI. Programación.

---

- ¿Qué características o propiedades tiene un Estudiante?

Estudiante

- a) Matrícula
- b) Carrera
- c) Nombre
- d) Apellido paterno
- e) Apellido materno

# Pilares de la TI. Programación.

---

- ¿Qué operaciones lleva a cabo un Estudiante?

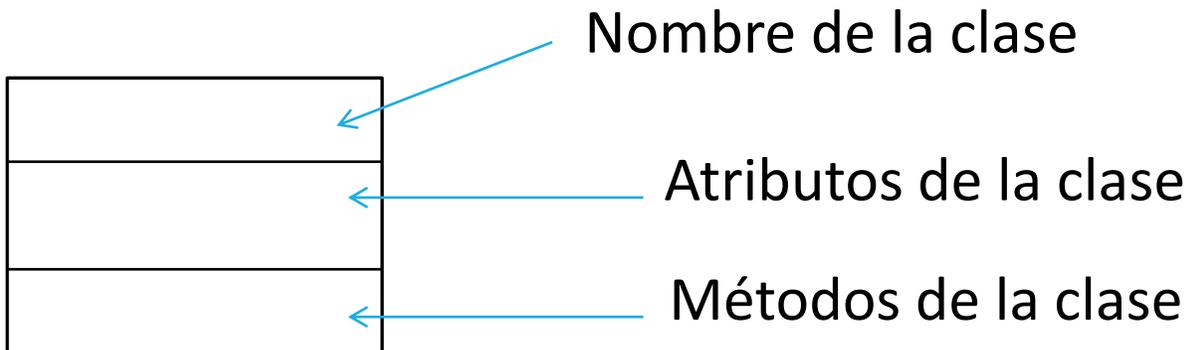
Estudiante

- a) Inscribirse
- b) Asistir a clases
- c) Estudiar

# Pilares de la TI. Programación.

---

Nuevamente... Qué elementos se incluyen en el modelado de la clase Estudiante:



<u>Estudiante</u>
Matrícula Carrera Nombre Apellido paterno Apellido materno
Inscribir( ) Asistir a clases( ) Estudiar( )

```
12 class Arbol {
13
14     // Clase que contiene cada nodo del arbol
15     class Nodo {
16     public:
17         int dato;
18         Nodo *izq;
19         Nodo *der;
20     };
21
22     // Variable para almacenar el nodo raiz del arbol
23     Nodo *raiz;
24
25     // Metodos de la clase
26
27     Nodo *crearNodo(int x)
28     {
29         Nodo *nuevo_nodo = new Nodo();
30         nuevo_nodo->dato = x;
31         nuevo_nodo->izq = NULL;
32         nuevo_nodo->der = NULL;
33
34         return nuevo_nodo;
35     }
36
37     Nodo *insertarNodo(int x, Nodo *arbol)
38     {
39         if(arbol == NULL)
40         {
```

# Pilares de la TI. Programación.

---

- Lenguajes orientados a objetos
- Modela un sistema como un conjunto de objetos cooperativos.
- Los objetos tienen atributos y comportamientos.
- Los objetos similares pueden agruparse y clasificarse en una clase.
- La interacción con los objetos se lleva a cabo a través de mensajes.
- Implica una nueva forma de ver las aplicaciones de negocios.
- Ejemplos: Smalltalk, C++, Java.

# Tecnologías de información, Pilares

---

# Gracias por su atención