

Programming Primer
Datos, abstracción y matemáticas

Programación de Estructuras de Datos y Algoritmos Fundamentales
(TC1031)

M.C. Xavier Sánchez Díaz
sax@tec.mx



Outline

1 Vocabulario

2 Datos

3 Abstracción de datos

4 Vocabulario avanzado

Términos comunes

Conceptos básicos de programación

- Estatuto (*Statement*)
- Función (*Function*)
- Rutina (*Routine*)
- Procedimiento (*Procedure*)
- Condicional (*Conditional*)
- Palabra reservada (*Reserved keyword*)
- Archivo (*File*)
- Directorio (*Directory*)
- Parámetro (*Parameter*)
- Argumento (*Argument*)
- Tipo (*Type*)
- Retorno (*Return*)
- Clase (*Class*)
- Operador (*Operator*)
- Iteración (*Iteration*)
- Variable (*Variable*)
- Estructura (*Structure*)
- Arreglo (*Array*)
- Objeto (*Object*)
- Bucle (*Loop*)
- Entero (*Integer*)
- Punto Flotante (*Floating-point*)
- Cadena de caracteres (*String*)
- Imprimir (*Print*)

Datos como resultados

Datos

Antes de usar la computadora o la calculadora para hacer cálculos, solíamos hacer las operaciones a mano.

Por ejemplo, si queremos calcular 1270×35 , una manera de hacerlo podría ser...

Datos como resultados

Datos

$$1270 \times 35 =$$

$$= (1200 + 70) \times (7)(5)$$

$$= (12)(7)(5)(100) + (7)(7)(5)(10)$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(100) + (7)(7)(5)(10)\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(100) + (7)(7)(5)(10)\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (\textcolor{red}{12})(7)(\textcolor{red}{5})(100) + (7)(7)(5)(10)\end{aligned}$$

$$\textcolor{red}{12} \times \textcolor{red}{5} = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(\textcolor{red}{7})(5)(100) + (7)(7)(5)(10)\end{aligned}$$

$$12 \times 5 = \textcolor{red}{60}$$

$$\textcolor{red}{60} \times \textcolor{red}{7} = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(\textcolor{red}{7})(5)(100) + (7)(7)(5)(10)\end{aligned}$$

$$12 \times 5 = \textcolor{red}{60}$$

$$\textcolor{red}{60} \times \textcolor{red}{7} = \textcolor{red}{6} \times \textcolor{red}{7} \times \textcolor{red}{10} = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(\textcolor{red}{100}) + (7)(7)(5)(10)\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = \textcolor{red}{420}$$

$$\textcolor{red}{420} \times \textcolor{red}{100} = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(100) + (\textcolor{red}{7})(\textcolor{red}{7})(5)(10)\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$\textcolor{red}{7} \times \textcolor{red}{7} = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(100) + (7)(7)(5)(\textcolor{red}{10})\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = \textcolor{red}{49}$$

$$\textcolor{red}{49} \times \textcolor{red}{10} = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(100) + (7)(7)(\textcolor{red}{5})(10)\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = \textcolor{red}{490}$$

$$\textcolor{red}{490} \times \textcolor{red}{5} = \textcolor{red}{490} \times \textcolor{red}{10/2} = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

$$\begin{aligned}1270 \times 35 &= \\&= (1200 + 70) \times (7)(5) \\&= (12)(7)(5)(100) + (7)(7)(5)(10)\end{aligned}$$

$$12 \times 5 = 60$$

$$60 \times 7 = 6 \times 7 \times 10 = 420$$

$$420 \times 100 = 42000$$

$$7 \times 7 = 49$$

$$49 \times 10 = 490$$

$$490 \times 5 = 490 \times 10/2 = 2450$$

$$42000 + 2450 = 44450 \quad \square$$

Datos como resultados

Datos

La operación completa se hace poco a poco, y por tanto necesitamos “recordar” ciertos pasos intermedios que ya tenemos calculados.

Así como nosotros tenemos que tener en claro cuáles son esos pasos intermedios, la computadora debe saber *dónde está* la información que tiene que leer para trabajar y hacer cálculos más elaborados.

Para eso, podemos usar las estructuras de datos, para **ordenarlos** de manera conveniente y poder tener acceso a ellos de manera que se vayan necesitando.

Datos como resultados

Datos

La operación completa se hace poco a poco, y por tanto necesitamos “recordar” ciertos pasos intermedios que ya tenemos calculados.

Así como nosotros tenemos que tener en claro cuáles son esos pasos intermedios, la computadora debe saber *dónde está* la información que tiene que leer para trabajar y hacer cálculos más elaborados.

Para eso, podemos usar las estructuras de datos, para **ordenarlos** de manera conveniente y poder tener acceso a ellos de manera que se vayan necesitando.

Datos como resultados

Datos

La operación completa se hace poco a poco, y por tanto necesitamos “recordar” ciertos pasos intermedios que ya tenemos calculados.

Así como nosotros tenemos que tener en claro cuáles son esos pasos intermedios, la computadora debe saber *dónde está* la información que tiene que leer para trabajar y hacer cálculos más elaborados.

Para eso, podemos usar las estructuras de datos, para **ordenarlos** de manera conveniente y poder tener acceso a ellos de manera que se vayan necesitando.

Datos y errores

Datos

¿Qué habría pasado si hubiera muchos puntos decimales de por medio?

¿Cuál es la representación decimal de $\frac{1}{7}$? ¿Y de $\frac{2}{7}$?

Story time: The Wolf

Datos y errores

Datos

¿Qué habría pasado si hubiera muchos puntos decimales de por medio?

¿Cuál es la representación decimal de $\frac{1}{7}$? ¿Y de $\frac{2}{7}$?

Story time: The Wolf

Datos y errores

Datos

¿Qué habría pasado si hubiera muchos puntos decimales de por medio?

¿Cuál es la representación decimal de $\frac{1}{7}$? ¿Y de $\frac{2}{7}$?

Story time: The Wolf

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

Tipos de datos

Datos

Existen distintos **tipos de datos** con los que podemos trabajar en una computadora:

- Números enteros (*integer numbers*)
- Números decimales (*floating point numbers*)
- Cadenas de caracteres alfanuméricos (*strings*)
- Datos estructurados definidos por nosotros mismos

Cuando **operamos** con estos datos, generamos nueva información que podríamos necesitar en el futuro.

¿Qué es un dato abstracto?

Abstracción de datos

Un **dato abstracto** es una manera *fancy* de decirle a una **estructura de datos** definida por el usuario.

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de datos

Asumamos que quiero saber las calificaciones de las Tareas 1, 2 y 3 de uno de mis alumnos. Para esto, necesitaría un lugar para guardar esos **3 datos**:

- $t_1 = 90$ será la variable para la Tarea 1
- $t_2 = 75$ será la variable para la Tarea 2
- $t_3 = 87$ será la variable para la Tarea 3

Con esta información, ahora contesta:

- ¿Cuál fue la calificación para la Tarea 2?
- ¿Cuál fue el promedio del alumno?
- ¿Cuál es la tarea en la que mejor le fue?

Arreglos

Abstracción de Datos

La pregunta ahora es...¿realmente necesito **3 variables** para guardar **3 datos**? Podemos *arreglar* los datos de tal manera que **su posición** nos aporte algo más:

$$t = \langle 90, 75, 87 \rangle$$

La **posición** en esta *estructura* nos indica qué número de tarea fue, y el valor que haya en dicha posición guarda la calificación. Por lo mismo, podemos usar “una sola variable” para guardar de manera estructurada la información requerida, y referirnos sólo a la posición deseada:

$$t_2 = 75$$

Arreglos

Abstracción de Datos

La pregunta ahora es...¿realmente necesito **3 variables** para guardar **3 datos**? Podemos *arreglar* los datos de tal manera que **su posición** nos aporte algo más:

$$t = \langle 90, 75, 87 \rangle$$

La **posición** en esta *estructura* nos indica qué número de tarea fue, y el valor que haya en dicha posición guarda la calificación. Por lo mismo, podemos usar “una sola variable” para guardar de manera estructurada la información requerida, y referirnos sólo a la posición deseada:

$$t_2 = 75$$

Arreglos

Abstracción de Datos

La pregunta ahora es... ¿realmente necesito **3 variables** para guardar **3 datos**? Podemos *arreglar* los datos de tal manera que **su posición** nos aporte algo más:

$$t = \langle 90, 75, 87 \rangle$$

La **posición** en esta *estructura* nos indica qué número de tarea fue, y el valor que haya en dicha posición guarda la calificación. Por lo mismo, podemos usar “una sola variable” para guardar de manera estructurada la información requerida, y referirnos sólo a la posición deseada:

$$t_2 = 75$$

Story time: La Torre Eiffel

Story time: La presentación del equipo 5

Story time: La Torre Eiffel

Story time: La presentación del equipo 5

Niveles de abstracción

Abstracción de Datos

Independientemente de la *estructura* de nuestros datos, a nivel abstracto siempre trabajamos con un **objeto**.

- Este objeto tiene varias cosas.
- Este objeto me sirve para otra cosa.
- Este objeto puedo usarlo de esta manera.

Esto es **abstracción**.

Fun fact: cambia el sustantivo “objeto” por *cosa* (o *madre*) para todo en esta slide y seguro lo recuerdas mejor

Niveles de abstracción

Abstracción de Datos

Independientemente de la *estructura* de nuestros datos, a nivel abstracto siempre trabajamos con un **objeto**.

- Este objeto tiene varias cosas.
- Este objeto me sirve para otra cosa.
- Este objeto puedo usarlo de esta manera.

Esto es **abstracción**.

Fun fact: cambia el sustantivo “objeto” por *cosa* (o *madre*) para todo en esta slide y seguro lo recuerdas mejor

Niveles de abstracción

Abstracción de Datos

Independientemente de la *estructura* de nuestros datos, a nivel abstracto siempre trabajamos con un **objeto**.

- Este objeto tiene varias cosas.
- Este objeto me sirve para otra cosa.
- Este objeto puedo usarlo de esta manera.

Esto es **abstracción**.

Fun fact: cambia el sustantivo “objeto” por *cosa* (o *madre*) para todo en esta slide y seguro lo recuerdas mejor

Niveles de abstracción

Abstracción de Datos

Independientemente de la *estructura* de nuestros datos, a nivel abstracto siempre trabajamos con un **objeto**.

- Este objeto tiene varias cosas.
- Este objeto me sirve para otra cosa.
- Este objeto puedo usarlo de esta manera.

Esto es **abstracción**.

Fun fact: cambia el sustantivo “objeto” por *cosa* (o *madre*) para todo en esta slide y seguro lo recuerdas mejor

Niveles de abstracción

Abstracción de Datos

Independientemente de la *estructura* de nuestros datos, a nivel abstracto siempre trabajamos con un **objeto**.

- Este **objeto** tiene varias **cosas**.
- Este **objeto** me sirve para **otra cosa**.
- Este **objeto** puedo usarlo de **esta manera**.

Esto es **abstracción**.

Fun fact: cambia el sustantivo “objeto” por *cosa* (o *madre*) para todo en esta slide y seguro lo recuerdas mejor

Niveles de abstracción

Abstracción de Datos

Independientemente de la *estructura* de nuestros datos, a nivel abstracto siempre trabajamos con un **objeto**.

- Este **objeto** tiene varias **cosas**.
- Este **objeto** me sirve para **otra cosa**.
- Este **objeto** puedo usarlo de **esta manera**.

Esto es **abstracción**.

Fun fact: cambia el sustantivo “objeto” por *cosa* (o *madre*) para todo en esta slide y seguro lo recuerdas mejor

Niveles de abstracción

Abstracción de Datos

- Una **clase** es un *tipo* de **objeto**
- Una clase tiene **atributos**
- Una clase tiene *operaciones* que puede realizar sobre sus **atributos**. Estas operaciones son **funciones** específicamente de la clase
- A estas **funciones** de la clase se les llama **métodos**
- Tanto los **métodos** como los **atributos** pueden ser **privados** o **públicos** dependiendo quién deba tener acceso a esa información.

Niveles de abstracción

Abstracción de Datos

- Una **clase** es un *tipo* de **objeto**
- Una **clase** tiene **atributos**
- Una **clase** tiene *operaciones* que puede realizar sobre sus **atributos**. Estas operaciones son **funciones** específicamente de la clase
- A estas **funciones** de la clase se les llama **métodos**
- Tanto los **métodos** como los **atributos** pueden ser **privados** o **públicos** dependiendo quién deba tener acceso a esa información.

Niveles de abstracción

Abstracción de Datos

- Una **clase** es un *tipo* de **objeto**
- Una **clase** tiene **atributos**
- Una **clase** tiene *operaciones* que puede realizar sobre sus **atributos**. Estas operaciones son **funciones** específicamente de la clase
- A estas **funciones** de la clase se les llama **métodos**
- Tanto los **métodos** como los **atributos** pueden ser **privados** o **públicos** dependiendo quién deba tener acceso a esa información.

Niveles de abstracción

Abstracción de Datos

- Una **clase** es un *tipo* de **objeto**
- Una **clase** tiene **atributos**
- Una **clase** tiene *operaciones* que puede realizar sobre sus **atributos**. Estas operaciones son **funciones** específicamente de la clase
- A estas **funciones** de la clase se les llama **métodos**
- Tanto los **métodos** como los **atributos** pueden ser **privados** o **públicos** dependiendo quién deba tener acceso a esa información.

Niveles de abstracción

Abstracción de Datos

- Una **clase** es un *tipo* de **objeto**
- Una **clase** tiene **atributos**
- Una **clase** tiene *operaciones* que puede realizar sobre sus **atributos**. Estas operaciones son **funciones** específicamente de la clase
- A estas **funciones** de la clase se les llama **métodos**
- Tanto los **métodos** como los **atributos** pueden ser **privados** o **públicos** dependiendo quién deba tener acceso a esa información.

Términos comunes

Conceptos básicos de programación orientada a objetos

- Clase (*Class*)
- Objeto (*Object*)
- Clase abstracta (*Abstract Class*)
- Interfaz (*Interface*)
- Instancia (*Instance*)
- Atributo (*Attribute*)
- Método (*Method*)
- Plantilla o Template (*Template*)
- Encapsulación (*Encapsulation*)
- Polimorfismo (*Polymorphism*)
- Herencia (*Inheritance*)
- Cohesión (*Cohesion*)
- Acoplamiento (*Coupling*)
- DRY
- *A function should do one thing*

Ejemplo en C++

Pa' recordarlo...

```
#include <iostream>
#include <vector>
#include <string>

using namespace std;

int main(){
    vector<string> msg {"Hello", "C++", "World", "from", "VS Code", "and the
↪ C++ extension!"};

    for (const string& word : msg){
        cout << word << " ";
    }
    cout << endl;
}
```