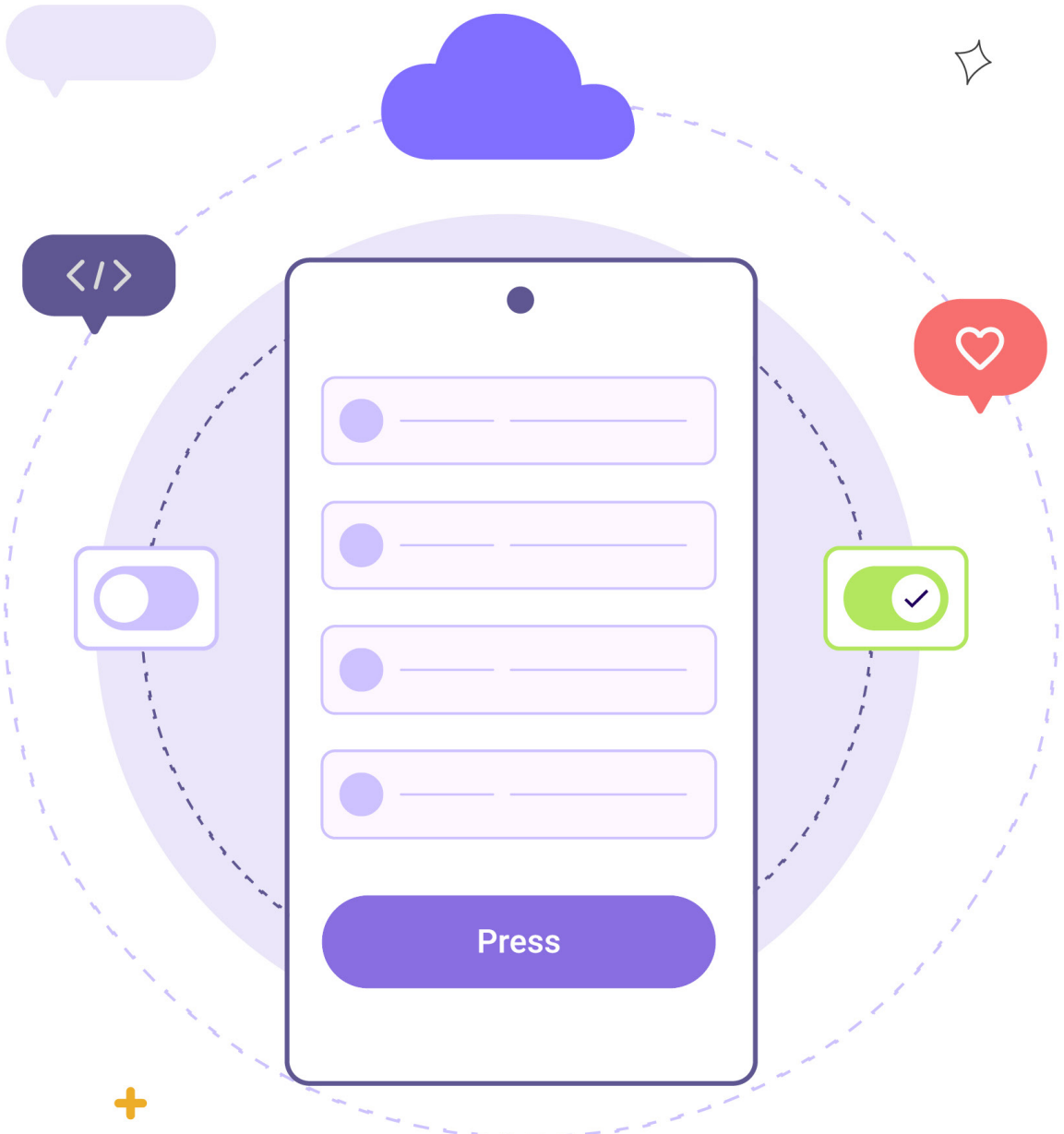
Capítulo 6

Estructuras de programa



Capítulo 6

Estructuras de programa

Objetivo

El objetivo de este capítulo es enseñar a organizar el código de forma efectiva mediante el uso de funciones y librerías. Aprenderemos a descomponer problemas complejos en subproblemas más pequeños y manejables, y a implementar soluciones utilizando funciones que encapsulan la lógica de cada subproblema. Además, exploraremos el uso de librerías para aprovechar código predefinido y ampliar las capacidades de nuestros programas.

Al finalizar este capítulo, estarás capacitado para estructurar tus programas. Habrás adquirido una comprensión profunda de los beneficios de las diferentes estructuras de programa disponibles y cómo utilizarlas para escribir código organizado, más comprensible y reutilizable.

Subproblemas

A medida que los programas comiencen a ganar complejidad, necesitaremos mayor cantidad de líneas de código, mayor cantidad de algoritmos y seguramente vamos a requerir utilizar más de una vez estos algoritmos.

Los subproblemas son en definitiva unidades más pequeñas e independientes que forman parte de un problema mayor. Al dividir un problema en subproblemas, podemos abordarlo de forma más sencilla y organizada.

La descomposición del problema original en subproblemas nos traerá varios beneficios, entre ellos destacamos:

- Simplifican la complejidad: dividen un problema grande en partes más manejables, facilitando su comprensión y resolución.
- Reutilización de código: permiten usar las mismas soluciones para diferentes problemas, ahorrando tiempo y esfuerzo.
- Mantenimiento: facilitan la detección y corrección de errores, al aislar el problema en un subcomponente específico.
- Legibilidad: hacen que el código sea más claro y fácil de leer, al agrupar la lógica en unidades independientes.

Los diferentes paradigmas de lenguajes proporcionan mecanismos para permitir estas abstracciones y facilitarnos el crecimiento de la complejidad en la búsqueda de soluciones algorítmicas.

El paradigma “Divide y vencerás” será nuestro aliado.

Este libro se encuentra dirigido a la enseñanza de la programación bajo el paradigma orientado a objetos y si bien existe un mecanismo específico para lograr esta abstracción por el momento utilizaremos el término función para referirnos a esta característica algorítmica.

Función

Llamaremos función a la encapsulación de bloques de código, la cual irá acompañada de un nombre, ejecutará un conjunto de acciones y retornará un determinado valor.

La utilización de funciones, en un lenguaje de programación nos dará varias ventajas, en particular haremos hincapié en la “reutilización”.

La reutilización de código será una herramienta esencial que tendremos como programadores, dado que nos permite analizar, diseñar e implementar ciertos algoritmos una única vez y luego podemos utilizarla las veces que se requiera.

Las funciones se encuentran caracterizadas por los siguientes elementos:

- Identificador: tiene un nombre único que las identifica y las diferencia de otras en un determinado contexto.
- Cuerpo: es el bloque de código que define su tarea.
- Retorno: pueden devolver o no un valor como resultado de su tarea.
- Parámetros: es la forma en la que podemos comunicarnos con la función. Son valores que la tarea necesita para su realización.

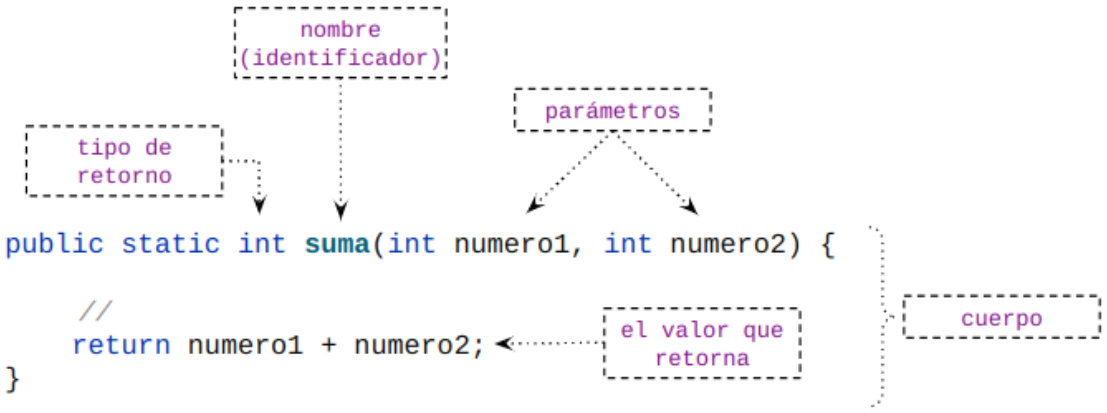


Fig.20 - partes de una función.

[-] identificador

Al igual que las variables, las funciones tendrán asociado un nombre único dentro del ámbito que se encuentren, concepto que profundizaremos más adelante.

Pueden existir funciones con el mismo nombre, siempre y cuando se encuentren en distintos ámbitos o sus parámetros sean diferentes en tipo o cantidad.

```
public static int suma(int a, int b){

    //
    return a + b;

}
```

[-] cuerpo

Es el bloque de código asociado a la función, entrará en ejecución cuando la función sea invocada.

El cuerpo de la función tendrá la dimensión que se necesite, dado que este dependerá de la tareas que se necesite realizar. Estas tareas pueden contar con variables locales, lo cual implica que su ámbito será dentro de la misma función.

Función que suma a y b: