

0101010
0100101
1101010

UD4.2.- DESCOMPOSICIÓ FUNCIONAL

0485 - Programació
1er DAW/DAW

0. CONTINGUTS

- **INTRODUCCIÓ**
- **DISSENY DESCENDENT**
- **AVANTATGES DEL DISSENY DESCENDENT**
- **ACTIVITAT PRÈVIA**

1. INTRODUCCIÓ

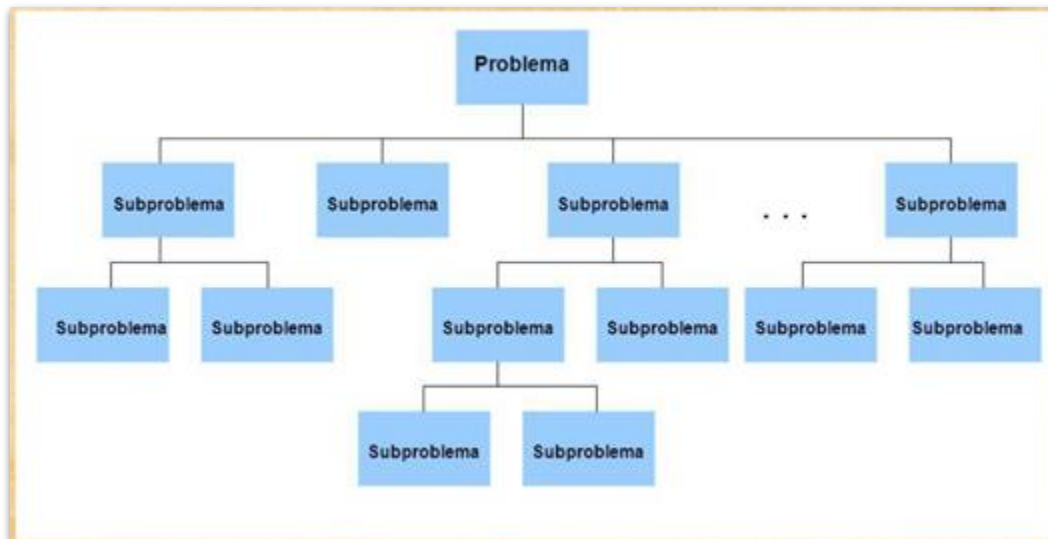
- Un **ERROR MOLT COMÚ** és tractar d'afrontar un problema **SENSE UN DISSENY PREVI** (directament codificant-lo).
 - L'ús de les **TIC** i la seua **integració a tots els àmbits** de la **vida quotidiana** (compres, transaccions bancàries, oci, ...) requereix de la implementació de **programes més complexos**.

Major complexitat
→
Major planificació



1.1 DISSENY DESCENDENT

- La **estratègia** a seguir és dividir el **problema principal** en **subproblemes** amb **diferents nivells de detall** (abstracció).
- La **solució** de cadascun dels **subproblemes individualment**, donarà lloc a la **resolució del problema general**.

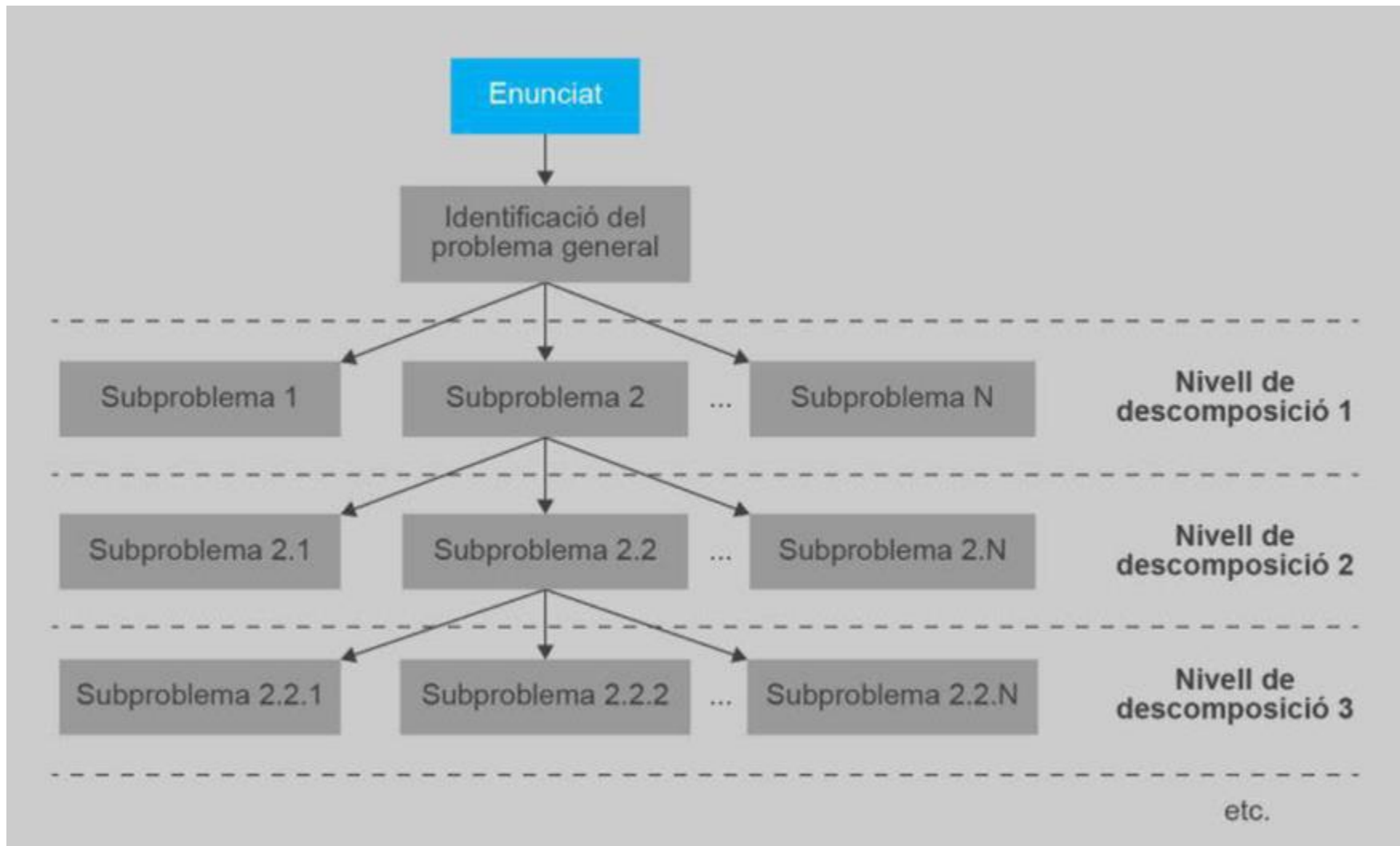


Per fer la
divisió
recorrem al
disseny
descendent

2. DISSENY DESCENDENT

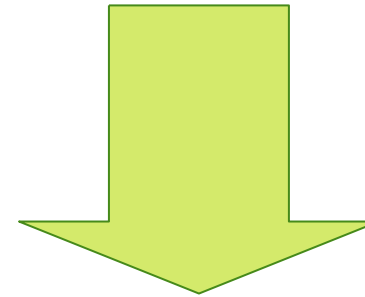
- La **descomposició del problema** no pot dur-se a terme de forma arbitrària.
 - Cada **subproblema** ha de **complir un objectiu parcial** i resoldre part del problema.
 - Es defineixen diferents **nivells d'abstracció**, cada nivell desenvolupa una funció i pot estar compost per diferents subproblemes.

2. DISSENY DESCENDENT



2.1 EXEMPLE I. RECEPТА

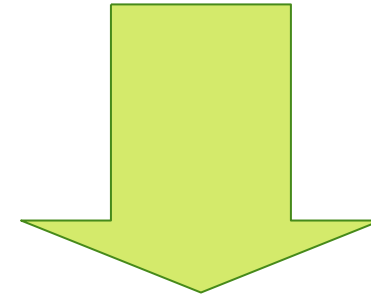
1. Preparació de fideus *yakisoba*.



- 1.1 Recopilar ingredients
- 1.2 Cuinar ingredients
- 1.3 Preparació Final

2.1 EXEMPLE I. RECEPТА

1.1 Recopilar ingredients



**1.1.1 Comprar al
supermercat.**

**1.1.2 Col·locar-los a la
taula.**

2.1 EXEMPLE I. RECEPTE

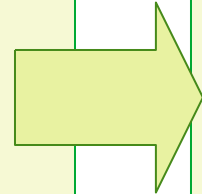
1.2 Cuinar ingredients

1.2 Cuinar ingredients

1.2.1 Cuinar tallarines

1.2.2 Cuinar Carlota

1.2.3 Cuinar Ceba



1.2.1.Cuinar tallarins

1.2.1.1 Bullir aigua.

1.2.1.2 Abocar els tallarins

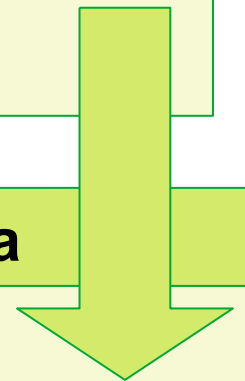
1.2.1.3 Escórrer els tallarins

1.2.1.4 Deixar-los en un plat

1.2.1.1 Bullir aigua

1.2.1.1.1 Preparar casso

1.2.1.1.2 Posar aigua



2.1 EXEMPLE I. RECEPТА

- El procés de ***Cuinar*** ingredients s'utilitzarà tant per als **tallarins** com per a les **cebes** i les **carlotes**.
- Els subproblemes de **nivells superiors** es basen en els **inferiors** per a **realitzar la seua tasca**.
- Anirem descomposant cada subproblema fins que es tracte de **tasques elementals o simples**.

2.2 EXEMPLE II. MENÚ

MENÚ APLICACIÓ

1. Veure Menú

1.1 Mostrar Opcions

1.1.1 Escriure en la eixida estàndar (pantalla)

Funció de la llibreria estàndar

1.2 Obtindre Selecció

1.2.1 Llegir de la entrada estàndar (teclat)

Funció de la llibreria estàndar

1.3 Manejar opció

2.2 EXEMPLE II. MENÚ

MENÚ APLICACIÓ

Nivell 1

Resol el problema de forma global
(Nivell de detall ALT)

1.veureMenu()

Nivell 2

Resol problemes concrets
(Nivell de detall BAIX)

1.1 mostrarOpcions()

1.2 obtindreSeleccio()

1.3 manejarOpcio()

2.2 EXEMPLE II. MENÚ

1.veureMenu()

```
public static void veureMenu() {  
  
    int opcionSeleccionada;  
    do {  
        mostrarOpciones();  
        opcionSeleccionada = obtindreSeleccio();  
        manejarOpcion(opcionSeleccionada);  
    } while (opcionSeleccionada != OPCION_SALIR);  
  
}
```

2.2 EXEMPLE II. MENÚ

1.2 obtindreSeleccio()

```
public static int obtindreSeleccio() {  
    do {  
        System.out.printf("Introduïxca una opció [%d - %d]: ", 1,  
NUM_OPCIONS);  
        if (teclado.hasNextInt()) {  
            return teclado.nextInt();  
        }  
        teclado.next();  
        System.out.println("Ha de seleccionar una opció vàlida");  
    } while (true);  
}
```

2.2 EXEMPLE II. MENÚ

1.1 mostrarOpciones()

```
private static void mostrarOpciones() {  
  
    System.out.printf("%d. Saludar %n", OPCIO_SALUDAR);  
    System.out.printf("%d. Comer %n", OPCIO_MENJAR);  
    System.out.printf("%d. Hablar %n", OPCIO_PARLAR);  
    System.out.printf("%d. Salir %n", OPCIO_EIXIR);  
  
}
```

2.2 EXEMPLE II. MENÚ

1.3 manejarOpcio()

```
private static void manejarOpcion(int opcionSeleccionada){  
    switch (opcionSeleccionada) {  
        case OPCION_SALUDAR:  
            System.out.println("Hola");  
            break;  
        case OPCION_COMER:  
            System.out.println("Menjat...");  
            break;  
        case OPCION_HABLAR:  
            System.out.println("En un lugar de la mancha...");  
            break;  
        case OPCION_SALIR:  
            System.out.println("Adéu...");  
            break;  
        default:  
            System.out.println("Ha de seleccionar una opción válida");  
    }  
}
```


ACTIVITAT PRÈVIA

- **Activitat 18.** Refactoritza el [següent exemple](#) fent ús de la tècnica de **descomposició funcional** vist a les diapositives prèvies.

```
public static void main(String[] args) {  
    Scanner teclado = new Scanner(System.in);  
    boolean salir = false;  
    do {  
        System.out.printf("%d. Saludar %n", OPCION_SALUDAR);  
        System.out.printf("%d. Comer %n", OPCION_COMER);  
        System.out.printf("%d. Hablar %n", OPCION_HABLAR);  
        System.out.printf("%d. Salir %n", OPCION_SALIR);  
        int opcionSeleccionada = teclado.nextInt();  
  
        switch (opcionSeleccionada) {  
            case OPCION_SALUDAR:  
                System.out.print("Hola a todos");  
                break;  
            case OPCION_COMER:  
                System.out.print("Comiendo");  
                break;  
            case OPCION_HABLAR:  
                System.out.print("En un lugar de la mancha....");  
                break;  
            case OPCION_SALIR:  
                System.out.print("Adios");  
                salir = true;  
                break;  
            default:  
                System.out.print("La opción seleccionada no es válida");  
        }  
    } while (!salir);  
}
```

2.3 AVANTATGES DEL DISSENY DESCENDENT.

- **Simplifica el programa;** fent fàcil llegir-lo i comprendre'l.
- Permet la **reutilització** de codi dins de cada subprograma.
- **Millora la mantenibilitat;** cada subproblema s'aïlla i es resol de forma individual facilitant el **desenvolupament**, la **depuració** i la **prova** del programa.
- Facilita el **treball en equip;** cada subproblema pot ser assignat a un membre de l'equip.

2.4 EXEMPLE III. ENDEVINA EL NÚMERO

Nivell 1

Resol el problema de forma global
(Nivell de detall ALT)

`jugarPartida()`

`veureIntroduccioJoc()`

`obtindreNumeroAdivinar()`

`obtindreIntentsUsuari()`

`ferIntents()`

`veureFinalPartida()`

Nivell 2

Resol subproblemes
(Nivell de detall MITJÀ)

Nivell 3

Resol subproblemes
(Nivell de detall MITJÀ)

`obtindreTiradaUsuari()`

`obtindreResultatTirada()`

Nivell 4

Resol problemes concrets
(Nivell de detall BAIX)

`generarEnterAleatori()`

`demanarEnterUsuari()`

`veurePista()`

2.4 EXEMPLE III. ENDEVINA EL NÚMERO

- Això és tot ... de moment :-)