**FUNDAMENTOS DE LA PROGRAMACION ORIENTADA A OBJETOS**

**Ejemplo 1**

**Definición del problema :** calcular la distancia entre la posición de Mario y la posición koopa.

**Análisis :**

**Datos de entrada :** posición Mario, posición Koopa.

**Datos de salida :** distancia entre Mario y Koopa.

**Proceso :** distanciaMK = distancia Koopa – distancia Mario.

**Diseño :**

**Entidad que resuelve el problema :** usuario

**Variables :**

Posición Mario, posición Koopa : enteros // almacenar datos al restar.

DistanciaMK : entero // almacenar datos de la resta.

**Nombre del algoritmo :** calcular\_distancia

**Proceso del algoritmo :**

* Leer posición Mario
* leer posición koopa
* DistanciaMK ← posición Koopa – posición Mario // restar y asignar el resultado
* Mostrar resultado de la resta

**Codificación :**

|  |  |
| --- | --- |
| **1** | **Int posicionM, posicionK, distanciaMK;** |
| **2** |  |
| **3** | **public void setup(){** |
| **4** | **posicionM=5;** |
| **5** | **posicionK=10;** |
| **6** | **calcularDistancia();** |
| **7** | **mostrarDistancia();** |
| **8** | **}** |
| **9** |  |
| **10** | **Public void calcularDistancia(){** |
| **11** | **distanciaMK=posicionK-posicionM** |
| **12** | **}** |
| **13** |  |
| **14** | **Public void mostrarDistancia(){** |
| **15** | **Println(distanciaMK)** |
| **16** | **}** |

**Awki Jarawi Gimenez Gutierrez**

**LU TUV000719**