

Comparación de Programación Imperativa vs Declarativa/Funcional

Javier Rosero

Enunciado

Dada una lista de estudiantes con sus respectivas calificaciones en un examen final, se requiere ordenarlos en **orden descendente según la nota**. En caso de empate, se ordenan **alfabéticamente por nombre**.

Se pide implementar la solución en dos paradigmas distintos: **imperativo** y **declarativo/funcional**. Posteriormente, realizar un análisis comparativo.

Solución Imperativa (Python)

```
1  # Paradigma imperativo
2  estudiantes = [("Ana", 85), ("Luis", 90), ("Carlos", 85),
3               , ("Marta", 95)]
4
5  # Ordenamiento paso a paso (burbuja modificada)
6  for i in range(len(estudiantes)):
7      for j in range(i + 1, len(estudiantes)):
8          if (estudiantes[j][1] > estudiantes[i][1]) or \
9              (estudiantes[j][1] == estudiantes[i][1] and
10               estudiantes[j][0] < estudiantes[i][0]):
11              estudiantes[i], estudiantes[j] = estudiantes
12              [j], estudiantes[i]
13
14  print("Ordenamiento_□(imperativo):")
15  for nombre, nota in estudiantes:
16      print(nombre, nota)
```

Listing 1: Código

Solución Declarativa/Funcional (Haskell)

```
1 import Data.List (sortBy)
2 import Data.Ord (comparing)
3
4 estudiantes :: [(String, Int)]
5 estudiantes = [("Ana",85), ("Luis",90), ("Carlos",85), ("Marta",95)]
6
7 -- Funcion de ordenamiento: primero nota (desc), luego
  nombre (asc)
8 ordenar :: [(String, Int)] -> [(String, Int)]
9 ordenar = sortBy (\(n1,c1) (n2,c2) -> compare c2 c1 <>
  compare n1 n2)
10
11 main :: IO ()
12 main = print (ordenar estudiantes)
```

Listing 2: Código

Este código compila correctamente con `ghc` y produce la lista ordenada como se espera.

Análisis Comparativo

- **Claridad y legibilidad:** El enfoque imperativo detalla paso a paso cómo ordenar, pero es más largo y difícil de leer. El declarativo expresa directamente el criterio de ordenamiento, siendo más conciso.
- **Nivel de abstracción:** El imperativo trabaja con bucles y estados mutables. El declarativo opera con funciones puras y mayor abstracción.
- **Mutabilidad e inmutabilidad:** Imperativo: usa mutabilidad (intercambio en listas). Declarativo: maneja estructuras inmutables.
- **Manejo de estado:** Imperativo: requiere variables de control. Declarativo: sin estado explícito, lo que reduce errores.
- **Mantenimiento y extensión:** Declarativo es más fácil de mantener y extender (solo se ajusta la función de orden). El imperativo exige cambios más profundos en el código.

- **Eficiencia:** Ambos pueden ser eficientes según el algoritmo. Imperativo puede ser más rápido en memoria (modificación en sitio). Declarativo es menos eficiente por inmutabilidad, pero más expresivo y seguro.