*Stefano Binotto*

*Ho eseguito l’homework2 V1*

Nel mio progetto ho utilizzato 3 diversi tipi di pattern:

* Class Pattern: ListAdapter, CollectionAdapter, MapAdapter.
* Object Pattern: SetAdapter
* Iterator Pattern: tutti gli Iterator e lisIterator

**CLASS ADAPTER:**

Questo tipo di pattern è stato utilizzato nelle classi ListAdapter, CollectionAdapter e MapAdapter per estendere i relativi Adaptee e implementare le relative interfacce. Questo pattern ha il vantaggio di essere più facile e veloce da sviluppare non dovendo dichiarare una istanza dell'Adaptee, tuttavia mi è risultato essere svantaggioso quando ho dovuto implementare Set, con un solo elemento, attraverso un hashtable con due elementi. Per questo motivo SetAdapter l'ho implementato utilizzando l'Object Adapter.

**OBJECT ADAPTER:**

Questo tipo di pattern è stato utilizzato nella classe SetAdapter. Poichè hastable contiene due elementi, mi dava problemi adattarlo ad un set che ne contiene solo uno. Usando questo tipo di Adapter questo problema è stato evitato. Inoltre lavorare con le istanze della classe che si vuole adattare rende la programmazione più flessibile.

**ITERATOR PATTERN:**

Questo tipo di pattern è stato utile per poter creare uno strumento che mi consentisse di fornire un oggetto per attraversare e modificare un contenitore. Nel caso della ListAdapter ho fornito due iteratori: Iterator e ListIterator. Questo pattern è molto vantaggioso in quanto consente di navigare e di apportare modifiche al contenitore. Tuttavia c'è uno svantaggio quando ci sono piu iteratori. Infatti la modifica di una collection da parte di un iteratore rende invalidi tutti gli altri iteratori usati.