

תיכון תוכנה ת.ב 2 – חלק יבש

זהו מימוש של Observable Monad:

```
class Observable<T: Listenable> private constructor(  
    private val observed: T,  
    private val observers: List<(Unit) -> Unit> = mutableListOf()  
) {  
    companion object {  
        fun <T:Listenable> of(observed: T, observers: List<(Unit) -> Unit> = listOf()): Observable<T> {  
            return Observable(observed, observers)  
        }  
    }  
  
    fun subscribe(f: (Unit) -> Unit): Observable<T> {  
        val newList = observers.toMutableList()  
        newList.add(f)  
        observed.whenDone(newList)  
        return Observable(observed, newList)  
    }  
  
    fun <S:Listenable> flatMap(f: (T, List<(Unit) -> Unit>) -> Observable<S>) = f(observed, observers)  
}
```

הסבר על המימוש:

Observable הינו אובייקט שיש עליו מאזינים (observers). המאזינים מיוצגים על ידי פונקציות ששמורות ברשימה. כעת, כאשר observable מסיים לחשב את הערך (לפי הממשק Listenable) הוא מבצע קריאה לכל הפונקציות שמייצגות את ה-observers.

נשים לב שבאופן מאוד טריוויאלי כללי ה-Monads מתקיימים עבור המימוש שהצגנו מכיוון ש הם שקולים לחלוטין למימוש של Box שגלמד בתרגול למעט העובדה שהצמדנו רשימה של observers שלא משתנה בין ההפעלות של הפונקציות שהכללים פועלים עליהם.