

Aufgabe 4: Lockfree**(9 Punkte)**

Gegeben sind folgende Klassen:

```
class SyncAccumulator {
    private int total = 0;
    private int lastValue = 0;

    public synchronized void update(int newValue) {
        total += newValue;
        lastValue = newValue;
    }

    public synchronized Result get() {
        return new Result(total, lastValue);
    }
}

class Result {
    public final int total;
    public final int lastValue;

    public Result(int total, int lastValue) {
        this.total = total; this.lastValue = lastValue;
    }
}
```

Mit dem Aufruf der Methode `update`, wird das Total nachgeführt und der letzte Wert, der zu diesem Total beigetragen hat, wird gespeichert. Die Methode `get` soll ein konsistentes `Result` liefern: Das Total, zusammen mit dem letzten Wert, der zu diesem Total geführt hat.

Aufgabe

Implementieren Sie die Klasse `LockFreeAccumulator`. Sie soll dieselbe Funktionalität anbieten wie die Klasse `SyncAccumulator` aber ohne Locks auskommen. Verwenden Sie dazu die Klassen aus dem Package `java.util.concurrent.atomic`.