

## OPTIMISTIC COVERAGE

Lo scopo di questo piccolo software è quello di registrare ogni giorno (ed eventualmente anche più volte al giorno) un nuovo file con una frase al suo interno, in modo da avere una collezione di frasi. E' importante mantenere TUTTE le frasi scritte.

Il funzionamento del software è molto semplice: c'è un solo metodo (`writeQuoteOfTheDay`) nel quale viene creato un nuovo file nella cartella `resources/quotes` e si scrive all'interno di tale file la stringa passata come parametro formale. Il test si occupa di verificare che questa scrittura avvenga correttamente in questo modo: a partire dal file descriptor contenuto all'interno della classe `QuoteOfTheDay` apre uno stream in lettura, legge la frase scritta e controlla che sia effettivamente uguale a quella passata per parametro. Il codice funziona correttamente e facendo un "run with coverage" si ottengono 0 errori con 100% coverage. In realtà però vi è un bug nel sistema, difficilmente individuabile dai test. Consideriamo di nuovo il metodo:

```
public void writeQuoteOfTheDay(String quote) throws IOException {
    DateTimeFormatter dtf = DateTimeFormatter.ofPattern("HH_mm_ss");
    LocalDateTime now = LocalDateTime.now();
    fd = new File("src/main/resources/quotes/file"+dtf.format(now));

    fd.createNewFile();
    BufferedWriter bw = new BufferedWriter(new FileWriter(fd));
    bw.write(quote);
    bw.close();
}
```

Quando si va a creare il nuovo file con la funzione `createNewFile` non si controlla il valore di ritorno che è molto importante: è vero se il file specificato non esiste e viene correttamente creato, falso se il file esiste. Dal momento che il nome del file viene scelto concatenando la stringa "file" al tempo corrente in ore, minuti, secondi potrebbe capitare che un giorno il file venga creato proprio nello stesso momento in cui ne era stato creato uno in precedenza. In questo caso andrei a sovrascrivere il contenuto precedente, comportamento non voluto dal sistema in quanto perderei la quote passata. Ovviamente il bug deriva da un errore logico nell'implementazione del metodo in cui si è pensato che concatenare il tempo al nome del file fosse sufficiente per garantirne l'unicità. Per risolvere il bug si deve usare la data corrente ed il tempo per creare il file piuttosto che solo il tempo oppure controllare il valore di ritorno di `createNewFile`.