

## プログラミング 2

### 課題レポート 3: リファクタリングを通してユニットテストとバージョン管理に慣れよう

225743C 赤嶺 紘基

2023 年 11 月 9 日

## 1 課題概要

### 課題概要

・コード例 ex-gradle2023 は、ターン制バトルゲーム（1 ターン毎に勇者とスライムが戦い、どちらかが倒れるまで攻撃し合う）を書いている途中だ。ただし次に示す問題があるため、修正したい。ユニットテストとバージョン管理システムを使いながら、一步步リファクタリングしてみよう。

**状況：**各ターンではまず勇者が敵を攻撃し、ダメージに応じてスライムの HP を修正する。次にスライムが勇者を攻撃し、勇者の HP を修正する。これを繰り返すが、どちらかが倒れたらループを抜ける。

**問題：**勇者がスライムを倒しても、スライムが勇者を攻撃できてしまう。

・問題を確認するテストコード (EnemyTest.java)。次の手順で問題の有無を確認している。

(1) ヒーローと敵を準備。ヒーローの攻撃力は敵を一撃で倒せるほど強い状態とする。具体的にはヒーローの攻撃力を 100 とし、スライムの HP を 10 とする。

(2) ヒーローが殴り、敵も殴る。敵は一撃で倒されていることを期待。実際には乱数を用いているためたまたま倒れないこともあるが、殆どの場合で即死ダメージを与える。

(3) 敵が死んだ状態ならば攻撃できないはず。つまり攻撃実行してもヒーローの HP は減っていないことを期待する。この期待通りになるかどうかをコードで検証する。

初期 HP を保存している変数 (defaultHeroHp) と、殴られた後のデモ勇者の HP (demoHero.hitPoint) とが等しいならば、減っていないと見做せる。

## 2 STEP1 について

push したりポジトリ URL (Git URL)

### 3 STEP2 について

指定手順で得られる `EnemyTest.attack()` の差分コードを掲載し、どのように修正したのか解説せよ。

### 4 STEP3 について

今回のコード修正を通して、気づいたことを報告せよ。

### 5 STEP4 について

キャプチャ画像を掲載せよ。