ASSIGNMENT 1 GUIDANCE

First, installing and setting up a Java Development Kit (JDK)

Compilation: `javac -cp class -d class source/*.java util/*.java`

- **`javac`:** This is the Java compiler command, used to compile .java source files into .class bytecode files.
- **`-cp class`** (or -classpath class)
    - The -cp (short for -classpath) option specifies the classpath, which tells the compiler where to look for existing .class files on which your Java code depends.
    - Here, class is the directory where precompiled .class files (e.g., Combatable.class) are stored. This allows your Java source files to reference these existing compiled classes.
- **`-d class`**
    - This specifies the output directory where the compiled .class files should be placed. In this case, class is the destination folder for compiled .class files.
- **`source/*.java util/*.java`:** This tells javac to compile all .java files in both the source and util directories (also add `Main.java` if `Main.java` has the same level of folder `source` and `util`).

Running the program: `java -cp class Main`

- `-cp class` → This tells Java to look for .class files inside the class/ directory.
- `Main` → This runs Main.class from class/.

Output:

```
nguyenminhtam@Tam-Nguyen Initial Code % javac -cp class -d class source/*.java util/*.java Main.java
nguyenminhtam@Tam-Nguyen Initial Code % java -cp class Main
  Moving to ground 849.
  Battle result. pR = 0.5927224362268005
```

In the `Battle` class, the variable `pr` accumulates and calculates the average probability of winning for Team 1 across all duels.

Pseudocode:

```
CLASS Battle:
    CONSTANT RATE_WIN = 0.5
    CONSTANT GROUND_BOUND = 999
    STATIC VARIABLE GROUND = 1  // Current battle ground
    METHOD moveRandomGround():
        Generate a random number between 1 and GROUND_BOUND
        Assign this number to GROUND
        Print "Moving to ground <GROUND>."
    METHOD constructor(team1, team2):
        SET mTeam1 = team1
        SET mTeam2 = team2
    METHOD combat():
        SET pr = 0  // Initialize total probability sum
        FOR each index i in range 0 to length of mTeam1:
            SET result = duel(mTeam1[i], mTeam2[i]) // Fight between two units
            ADD result to pr  // Accumulate win probability
            // Special rule: If first fighter in Team 1 wins, change ground
            IF i == 0 AND result >= RATE_WIN:
```

```
        CALL moveRandomGround()
    // Compute the average probability of Team 1 winning
    pr = pr / length of mTeam1
    Print "Battle result. pR = <pr>"
METHOD duel(cb1, cb2):
    SET score1 = cb1.getCombatScore()
    SET score2 = cb2.getCombatScore()
    // Compute probability of Team 1's fighter winning
    pr = (score1 - score2 + 999) / 2000.0
    RETURN pr
```