

See the Assessment Guide for information on how to interpret this report.

ASSESSMENT SUMMARY

Compilation: PASSED
API: PASSED

SpotBugs: PASSED
PMD: PASSED
Checkstyle: FAILED (0 errors, 1 warning)

Correctness: 23/23 tests passed
Memory: 4/4 tests passed
Timing: 1/1 tests passed

Aggregate score: 100.00%
[Compilation: 5%, API: 5%, Style: 0%, Correctness: 60%, Timing: 10%, Memory: 20%]

ASSESSMENT DETAILS

The following files were submitted:

6.6K Nov 27 07:22 BaseballElimination.java

* COMPILING

% javac BaseballElimination.java
*-----

=====

Checking the APIs of your programs.
*-----
BaseballElimination:

=====

* CHECKING STYLE AND COMMON BUG PATTERNS

% spotbugs *.class
*-----

=====

% pmd .
*-----

=====

% checkstyle *.java

```
-----
% custom checkstyle checks for BaseballElimination.java
*-----
[WARN] BaseballElimination.java:12:52: The numeric literal '4' appears to be unnecessary. [NumericLiteral]
Checkstyle ends with 0 errors and 1 warning.

=====

*****
*   TESTING CORRECTNESS
*****

Testing correctness of BaseballElimination
*-----
Running 23 total tests.

Test 1: check numberOfTeams()
  * teams4.txt
  * teams5.txt
  * teams8.txt
  * teams10.txt
  * teams29.txt
  * teams48.txt
==> passed

Test 2: check teams()
  * teams4.txt
  * teams5.txt
  * teams8.txt
  * teams10.txt
  * teams29.txt
  * teams48.txt
==> passed

Test 3: check wins()
  * teams4.txt
  * teams5.txt
  * teams8.txt
  * teams10.txt
  * teams29.txt
  * teams48.txt
==> passed

Test 4: check losses()
  * teams4.txt
  * teams5.txt
  * teams8.txt
  * teams10.txt
  * teams29.txt
  * teams48.txt
==> passed

Test 5: check remaining()
  * teams4.txt
  * teams5.txt
  * teams8.txt
  * teams10.txt
  * teams29.txt
  * teams48.txt
==> passed

Test 6: check against()
  * teams4.txt
  * teams5.txt
  * teams8.txt
  * teams10.txt
  * teams29.txt
  * teams48.txt
==> passed

Test 7a: check isEliminated() when n = 4
```

```
* teams4.txt
* teams4a.txt
* teams4b.txt
==> passed
```

```
Test 7b: check isEliminated() when n = 5
* teams5.txt
* teams5a.txt
* teams5b.txt
* teams5c.txt
==> passed
```

```
Test 7c: check isEliminated() when n = 6 to 19
* teams7.txt
* teams8.txt
* teams10.txt
* teams12.txt
* teams12-allgames.txt
==> passed
```

```
Test 7d: check isEliminated() when n = 20 to 30
* teams24.txt
* teams29.txt
* teams30.txt
==> passed
```

```
Test 7e: check isEliminated() when n = 30 to 50
* teams32.txt
* teams36.txt
* teams42.txt
* teams48.txt
==> passed
```

```
Test 8: check that isEliminated() is consistent with certificateOfElimination()
* teams4.txt
* teams5.txt
* teams8.txt
* teams10.txt
* teams29.txt
* teams48.txt
==> passed
```

```
Test 9a: check certificateOfElimination() when n = 4
* teams4.txt
* teams4a.txt
* teams4b.txt
==> passed
```

```
Test 9b: check certificateOfElimination() when n = 5
* teams5.txt
* teams5a.txt
* teams5b.txt
* teams5c.txt
==> passed
```

```
Test 9c: check certificateOfElimination() when n = 6 to 20
* teams7.txt
* teams8.txt
* teams10.txt
* teams12.txt
* teams12-allgames.txt
==> passed
```

```
Test 9d: check certificateOfElimination() when n = 20 to 30
* teams24.txt
* teams29.txt
* teams30.txt
==> passed
```

```
Test 9e: check certificateOfElimination() when n = 30 to 50
* teams32.txt
* teams36.txt
* teams42.txt
* teams48.txt
```

==> passed

Test 10: check methods when n = 1

- * teams1.txt
- * teams1.txt
- * teams1.txt
- * teams1.txt
- * teams1.txt
- * teams1.txt
- * teams1.txt

==> passed

Test 11: check immutability by calling certificateOfElimination() both before and after calling isEliminated() and verifying that the results match

- * teams4.txt
- * teams5.txt
- * teams29.txt

==> passed

Test 12: check that certificateOfElimination() returns null for eliminated teams

- * teams4.txt
- * teams5.txt

==> passed

Test 13: check for invalid arguments

- * teams4.txt
- * teams5.txt

==> passed

Test 14: check for dependence on reference equality of strings

==> passed

Test 15: check that two Baseball objects can be created at the same time

- * teams4.txt and teams5.txt
- * teams5.txt and teams7.txt

==> passed

Total: 23/23 tests passed!

=====

* TIMING

Timing BaseballElimination

*-----

Running 1 total tests.

n	constructor	isEliminated() + certificateOfElimination()
30	0.01	0.47
36	0.00	0.99
42	0.00	1.95
48	0.00	3.63
54	0.00	6.12
60	0.00	10.23

time = 1.33e-07 * n^4.43 (R^2 = 1.00)

Total: 1/1 tests passed!

=====

* MEMORY

Analyzing memory of BaseballElimination

*-----

Running 4 total tests.

Student vertices = $0.50 n^2 - 0.50 n + 3.00$ ($R^2 = 1.000$)
Reference vertices = $0.50 n^2 - 0.50 n + 3.00$ ($R^2 = 1.000$)
=> passed

Student edges = $1.50 n^2 - 3.50 n + 3.00$ ($R^2 = 1.000$)
Reference edges = $1.50 n^2 - 3.50 n + 2.00$ ($R^2 = 1.000$)
=> passed

Student memory of G = $176.00 n^2 - 384.00 n + 488.00$ ($R^2 = 1.000$)
Reference memory of G = $176.00 n^2 - 384.00 n + 384.00$ ($R^2 = 1.000$)
=> passed

Student memory = $3.83 n^2 + 200.32 n + 214.86$ ($R^2 = 1.000$)
Reference memory = $3.99 n^2 + 214.25 n + 396.00$ ($R^2 = 1.000$)
=> passed

Total: 4/4 tests passed!

=====