

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, 2 warnings)

Correctness: 13/13 tests passed  
 Memory: No tests available for autograding.  
 Timing: No tests available for autograding.

Aggregate score: 100.00%  
 [ Compilation: 5%, API: 5%, Style: 0%, Correctness: 90% ]

## ASSESSMENT DETAILS

The following files were submitted:

-----  
 500 Sep 13 09:48 HelloGoodbye.java  
 392 Sep 13 09:48 HelloWorld.java  
 812 Sep 13 09:48 RandomWord.java

\*\*\*\*\*  
 \* COMPILING  
 \*\*\*\*\*

% javac HelloWorld.java  
 \*-----

% javac HelloGoodbye.java  
 \*-----

% javac RandomWord.java  
 \*-----

=====

Checking the APIs of your programs.  
 \*-----

HelloWorld:

HelloGoodbye:

RandomWord:

=====

\*\*\*\*\*  
 \* CHECKING STYLE AND COMMON BUG PATTERNS  
 \*\*\*\*\*

% spotbugs \*.class  
 \*-----

=====

% pmd .  
 \*-----

=====

% checkstyle \*.java  
 \*-----

```
% custom checkstyle checks for HelloWorld.java
*-----

% custom checkstyle checks for HelloGoodbye.java
*-----

% custom checkstyle checks for RandomWord.java
*-----
[WARN] RandomWord.java:20: Calling 'StdIn.isEmpty()' in more than one place suggests poor design in this program. [Design]
[WARN] RandomWord.java:21: Calling 'StdIn.readString()' in more than one place suggests poor design in this program. [Design]
Checkstyle ends with 0 errors and 2 warnings.
```

```
=====
```

```
*****
* TESTING CORRECTNESS
*****
```

```
Testing correctness of HelloWorld
```

```
*-----
```

```
Running 2 total tests.
```

```
Test 1: check output format
```

```
% java HelloWorld
Hello, World
```

```
==> passed
```

```
Test 2: check correctness
```

```
* java HelloWorld
```

```
==> passed
```

```
HelloWorld Total: 2/2 tests passed!
```

```
=====
```

```
Testing correctness of HelloGoodbye
```

```
*-----
```

```
Running 6 total tests.
```

```
Test 1: check output format
```

```
% java HelloGoodbye Kevin Bob
Hello Kevin and Bob.
Goodbye Bob and Kevin.
```

```
% java HelloGoodbye Alejandra Bahati
Hello Alejandra and Bahati.
Goodbye Bahati and Alejandra.
```

```
==> passed
```

```
Test 2: check correctness using names from assignment specification
```

```
* java HelloGoodbye Kevin Bob
* java HelloGoodbye Alejandra Bahati
```

```
==> passed
```

```
Test 3: check correctness using fixed names
```

```
* java HelloGoodbye Chandra Deshi
* java HelloGoodbye Ayşe Maria
* java HelloGoodbye Wayan Taiyeo
* java HelloGoodbye Ástfríður Bedřiška
```

```
==> passed
```

```
Test 4: check correctness when two names are the same
```

```
* java HelloGoodbye Turing Turing
* java HelloGoodbye Lovelace Lovelace
* java HelloGoodbye Hopper Hopper
* java HelloGoodbye Knuth Knuth
```

```
==> passed
```

```
Test 5: check correctness using random names
```

```
* java HelloGoodbye Ameetha Shaniqua
* java HelloGoodbye Mertcan Silia
* java HelloGoodbye Varo Jessy
* java HelloGoodbye Horatio Stieg
* java HelloGoodbye Micaela Münür
```

```
==> passed
```

```
Test 6: test correctness using many random names
```

```
* 10 pairs of random names
* 100 pairs of random names
* 1000 pairs of random names
* 10000 pairs of random names
```

```
==> passed
```

HelloGoodbye Total: 6/6 tests passed!

=====  
Testing correctness of RandomWord

\*-----  
Running 5 total tests.

Test 1: check that programs prints exactly one word

% java-algs4 RandomWord < animals8.txt  
horse

% java-algs4 RandomWord < animals8.txt  
fox

% java-algs4 RandomWord < coin.txt  
heads

% java-algs4 RandomWord < names.txt  
Hui+Qi

Test 2: check that the word the program prints appears on standard input

\* 10000 trials from coin.txt  
\* 10000 trials from animals8.txt  
\* 10000 trials from animals.txt  
\* 10000 trials from colors.txt  
\* 10000 trials from algorithms.txt  
\* 10000 trials from dream.txt

Test 3: check that program consumes all data from standard input

\* coin.txt  
\* animals8.txt  
\* animals.txt  
\* colors.txt  
\* algorithms.txt  
\* names.txt  
\* dream.txt

Test 4: check that words printed are uniformly random

\* 12000 trials from coin.txt  
\* 12000 trials from roshambo.txt  
\* 12000 trials from die.txt  
\* 12000 trials from animals8.txt  
\* 12000 trials from animals25.txt  
\* 12000 trials from algorithms.txt

Test 5: check that words printed are uniformly random  
(with some words on standard input repeated)

\* 36000 trials from biased-coin.txt  
\* 36000 trials from loaded-die.txt  
\* 36000 trials from two-dice.txt  
\* 36000 trials from tinyTale.txt

RandomWord Total: 5/5 tests passed!

=====