



Course Progress

Course > Readings/Videos > Reading 4: Specifications > Questions

 Previous



























Next 

Questions

 Bookmark this page

words with specs

2/2 points (graded)

Suppose we're working on the method below:

```
/**
 * Requires: tiles has length 7 & contains only uppercase letters.
 *           crossings contains only uppercase letters, without duplicates.
 * Effects: Returns a list of words where each word can be made by taking
 *          letters from tiles and at most 1 letter from crossings.
 */
public static List<String> scrabble(String tiles, String crossings) {
    if (tiles.length() != 7) { throw new RuntimeException(); }
    return new ArrayList<>();
}
```

Which are parts of the *postcondition* of `scrabble` ?

☐ `tiles` has only uppercase letters

☐ `crossings` has no duplicates

☐ `scrabble` takes two arguments

☒ `scrabble` returns a list of strings



Explanation

The first three choices are preconditions. Only the last choice is a postcondition.

Which are parts of the *precondition* of `scrabble` ?

☒ `tiles` has length 7

☒ `crossings` is a string of uppercase letters

☒ `scrabble` 's arguments are of type `String` and `String`

☐ `scrabble` returns an empty `ArrayList`



Explanation

The last answer is true about the implementation, but it is not part of the spec.

Submit

Show Answer

Answers are displayed within the problem

Scrabble scramble

2/2 points (graded)

Which are *parts of the spec that are checked statically* by Java?

☐ `tiles` is a string of uppercase letters

☐ `crossings` has no duplicates

☐ when `tiles.length() != 7`, `scrabble` throws a `RuntimeException`

☒ `scrabble` takes two arguments



Does the `scrabble` implementation satisfy its specification?

☒ Yes

☐ No, because it throws a `RuntimeException` when `tiles` has invalid length

☐ No, because it returns an empty result list even if we call it with tiles and crossings that can be combined to form valid words



Explanation

The specification doesn't say that the method returns *all* words that can be made from the inputs, or even *some* words that can be made. This is a clear place where the specification could be improved.

Submit

Show Answer

Answers are displayed within the problem

◀ Previous

Next ▶

Some Rights Reserved

[Open Learning Library](#)

[About](#)

[Accessibility](#)

[All Courses](#)

[Why Support MIT Open Learning?](#)

[Help](#)

[Connect](#)

[Contact](#)

[Twitter](#)

[Facebook](#)

[Privacy Policy](#) [Terms of Service](#)

© Massachusetts Institute of Technology, 2024