

Revision: category-partition method (2008 exam, question 1)

Conrad Hughes

March 23, 2009

The `String.split()` method was added to the Java specification at version 1.4, and is the preferred modern replacement for the legacy `StringTokenizer` class:

```
// class String
public String[] split(String regex, int limit)
```

`split()` breaks *this* `String` into an array of fragments delimited by matches of the regular expression *regex*. No more than *limit* fragments are returned (the `/*i*/` comments here are shown purely to highlight how the method operates):

```
s = "mississippi"
s.split("i", 999) -> { "m", "ss", "ss", "pp", "" }
                      /*i*/ /*i*/ /*i*/ /*i*/
```

Effect of *limit*:

- When *limit* is positive, if there are more than $(limit - 1)$ matches of *regex* in the string (and hence more than *limit* fragments), then the array will contain exactly *limit* elements, and the last element will contain the unsplit excess from the string (*s* here has the same value as above):

```
s.split("i", 3) -> { "m", "ss", "ssippi" }
```

- When *limit* is negative, there is no limit on the number of fragments returned:

```
s.split("i", -1) -> { "m", "ss", "ss", "pp", "" }
```

- When *limit* is zero, there is also no limit on the number of fragments returned, **but** all empty **trailing** fragments are elided:

```
s.split("i", 0) -> { "m", "ss", "ss", "pp" }
s.split("iss", 0) -> { "m", "", "ippi" }
```

1. Using just the information provided here, what test methodology would you apply to an implementation of `String.split()`? Justify your answer. What extra information about the implementation would be useful for testing, and how might access to this information affect the testing process? Suggest reasons why this might improve **or** **impair** the testing process.
2. Work through the test method you have suggested, in order to generate a test specification for `String.split()`. Make sure that you clearly identify all steps involved in the process as you apply them. Do not use the string “mississippi” in your test specification.

Note:

- You should assume that the regular expression features of `split()` are to be tested by someone else, so for this question *regex* is a simple string (as in the examples above).
 - If the specification appears ambiguous or inadequate, note this, and make (and declare) any assumptions you feel necessary to allow you to continue with your work.
3. When testing systems with many parameters, we often initially identify an impractically large number of possible tests. Describe, in reasonable detail, the operation of **two** techniques by which the number of tests can be reduced. Make reference to your answer to part (b) or provide examples.