Ken Steinfeldt
Dennie Devito
Final Project Part A
5/22/17

Explain testIsValid function

The testIsValid function takes an object array of URL parts and appends the parts in order to create a full URL. Once a full URL has been formed, that URL is validated by the URL validated and its result is compared to an expected value. If these values are equal, then the URL validator is working correctly, otherwise, it is not, and the test will fail. It should be noted that this function runs through every possible permutation possible given the URL parts it is given. If a single part of the URL is not valid, the entire URL is not valid.

How many total number of URLs

The test suite breaks a URL down into four parts; scheme, authority, path, and query. Moreover another part is added, port, in order to create more possible permutations. The total number of permutations can be found my multiplying the number of options in each part. Scheme[9]*authority[19]*port[7]*path[10]*query[3] = 35910 possible permutations. There is also an additional path option, the allowance of 2 slashes, that results in 15 more options for a total of 538650 possible permutations.

Example of valid
A valid URL example: http://www.google.com:80/test1?action=view

Example of invalid
An invalid URL example: http:www.google.com:-1/test1?action=view

How different is this testing from ours in class?

This test is slightly more sophisticated than what we have done in class and also creates a fairly vast array of possible permutations to be tested. Testing over 35 thousand permutations is much different than anything that we have written in class. The test function is also quite robust in its own right, as it builds the permutations by iterating over several object arrays, and can test several different options available to the function. However, practically speaking, it really is not that different from what we have written in class. The test asserts that the return value equals that of the expected value, and if it does not, the test will fail.