

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 // O(n + m) time | O(m) space
4 function knuthMorrisPrattAlgorithm(string, substring) {
5   let pattern = buildPattern(substring);
6   return doesMatch(string, substring, pattern);
7 }
8
9 function buildPattern(substring) {
10  let pattern = new Array(substring.length).fill(-1);
11  let j = 0;
12  let i = 1;
13  while (i < substring.length) {
14    if (substring[i] === substring[j]) {
15      pattern[i] = j;
16      i++;
17      j++;
18    } else if (j > 0) {
19      j = pattern[j - 1] + 1;
20    } else {
21      i++;
22    }
23  }
24  return pattern;
25 }
26
27 function doesMatch(string, substring, pattern) {
28  let i = 0;
29  let j = 0;
30  while (i + substring.length - j <= string.length) {
31    if (string[i] === substring[j]) {
32      if (j === substring.length - 1) return true;
33      i++;
34      j++;
35    } else if (j > 0) {
36      j = pattern[j - 1] + 1;
37    } else {
38      i++;
39    }
40  }
41  return false;
42 }
43
44 exports.knuthMorrisPrattAlgorithm = knuthMorrisPrattAlgorithm;
45
```

