## Input:

(For creation) : String = "babc"

String = "abc"

## Output:

(For creation): {
    "c" : { "*": true },
    "b" : {
      "c": { "*": true }
      "a": { "b": { "c": {"*": true}}},
    },
    "a": {"b" : {"c" : { "*": true }}},
}

true

```
class SuffixTrie {
  constructor(string) {
    this.root = {};
    this.endSymbol = '*';
    this.populateSuffixTrieFrom(string);
  }

  // O(n^2) time | O(n^2) space
  populateSuffixTrieFrom(string) {
    for (let i = 0; i < string.length; i++) {
      this.insertSubstringStartingAt(i, string);
    }
  }

  insertSubstringStartingAt(i, string) {
    let node = this.root;
    for (let j = i; j < string.length; j++) {
      const letter = string[j];
      if (!(letter in node)) node[letter] = {};
      node = node[letter];
    }
    node[this.endSymbol] = true;
  }

  // O(m) time | O(1) space
  contains(string) {
    let node = this.root;
    for (const letter of string) {
      if (!(letter in node)) return false;
      node = node[letter];
    }
    return this.endSymbol in node;
  }
}
```