



Tecnológico de Monterrey

Tecnológico de Monterrey - Campus Monterrey
School of Engineering and Sciences
Engineering in Computational Technologies
Analysis and Design of Advanced Algorithms

Homework 9: Suffix Trie

Group: 607
Team #3

Luis Salomón Flores Ugalde

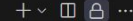
Santiago Quintana Moreno A01571222
Miguel Ángel Álvarez Hermida a01722925



HW9_SuffixTrie.py X



powershell X



Homework9_SuffixTrie > HW9_SuffixTrie.py > TrieNode > _init_

```
1 # Analysis and Design of Advanced Algorithms
2 # Group #607
3 # Team 3
4 # Luis Salomón Flores Ugalde
5
6 # Santiago Quintana Moreno A01571222
7 # Miguel Ángel Álvarez Hermida A01722925
8
9 # ----- SUFFIX TIRE -----
```

```
10
11 class TrieNode:
12     __slots__ = ("children", "is_end")
13     def __init__(self):
14         self.children = {}
15         self.is_end = False
16
17 class Trie:
18     def __init__(self, words=None):
19         self.root = TrieNode()
20         if words:
21             for w in words:
22                 self.insert(w)
23
24     def insert(self, word: str) -> None:
25         node = self.root
26         for ch in word:
27             if ch not in node.children:
28                 node.children[ch] = TrieNode()
29             node = node.children[ch]
30         node.is_end = True
31
32     def search(self, word: str) -> bool:
33         node = self.root
34         for ch in word:
35             if ch not in node.children:
36                 return False
37             node = node.children[ch]
38         return node.is_end
39
40     def starts_with(self, prefix: str) -> bool:
41         node = self.root
42         for ch in prefix:
```

```
PS D:\1.SQM\1.UNIVERSIDAD\5. QUINTO SEMESTRE\2.Advanced Algorithms> & C:\Users\santy
\AppData\Local\Microsoft\WindowsApps\python3.13.exe "d:/1.SQM/1.UNIVERSIDAD/5. QUIPS
0 SEMESTRE/2.Advanced Algorithms/Homework9_SuffixTrie/Hw9_SuffixTrie.py"
```

Dictionary Trie tests:

```
'Fire': FOUND
'Fira': FOUND
'Firaga': FOUND
'Firaja': FOUND
'Firas': NOT FOUND
'Fi': NOT FOUND
```

Suffix Trie pattern searches in text: anabanana

```
'ana' -> indices [0, 4, 6]
'nana' -> indices [5]
'ban' -> indices [3]
'x' -> indices []
'a' -> indices [0, 2, 4, 6, 8]
'anabanana' -> indices [0]
```

PS D:\1.SQM\1.UNIVERSIDAD\5. QUINTO SEMESTRE\2.Advanced Algorithms>

FileEditSelectionViewGoRunTerminalHelp←→2.Advanced Algorithms

HW9_SuffixTrie.py ×

Homework9_SuffixTrie > HW9_SuffixTrie.py > TrieNode > __init__

```
17 class Trie:
39
40     def starts_with(self, prefix: str) -> bool:
41         node = self.root
42         for ch in prefix:
43             if ch not in node.children:
44                 return False
45             node = node.children[ch]
46         return True
47
48
49 class SuffixTrieNode:
50     __slots__ = ("children", "indices")
51     def __init__(self):
52         self.children = {}
53         self.indices = []
54
55
56 class SuffixTrie:
57     def __init__(self, text: str):
58         self.text = text
59         self.root = SuffixTrieNode()
60         self._build()
61
62     def _build(self):
63         t = self.text
64         n = len(t)
65         for i in range(n):
66             node = self.root
67             node.indices.append(i)
68             for ch in t[i:]:
69                 if ch not in node.children:
70                     node.children[ch] = SuffixTrieNode()
71                 node = node.children[ch]
72             node.indices.append(i)
73
74     def occurrences(self, pattern: str):
75         if not pattern:
76             return []
77         node = self.root
78         for ch in pattern:
79             if ch not in node.children:
```

powershell ×

PS D:\1.SQM\1.UNIVERSIDAD\5. QUINTO SEMESTRE\2.Advanced Algorithms> & C:\Users\santy\AppData\Local\Microsoft\WindowsApps\python3.13.exe "d:/1.SQM/1.UNIVERSIDAD/5. QUIPS 0 SEMESTRE/2.Advanced Algorithms/Homework9_SuffixTrie/Hw9_SuffixTrie.py"

Dictionary Trie tests:

'Fire': FOUND

'Fira': FOUND

'Firaga': FOUND

'Firaja': FOUND

'Firas': NOT FOUND

'Fi': NOT FOUND

Suffix Trie pattern searches in text: anabanana

'ana' -> indices [0, 4, 6]

'nana' -> indices [5]

'ban' -> indices [3]

'x' -> indices []

○ 'a' -> indices [0, 2, 4, 6, 8]

'anabanana' -> indices [0]

PS D:\1.SQM\1.UNIVERSIDAD\5. QUINTO SEMESTRE\2.Advanced Algorithms> []

main ↺ 🔍 0 ⚠ 0

Go Live 🔔

FileEditSelectionViewGoRunTerminalHelp←→2.Advanced Algorithms

HW9_SuffixTrie.py ×

HW9_SuffixTrie.py > TrieNode > __init__

```
56 class SuffixTrie:
73
74     def occurrences(self, pattern: str):
75         if not pattern:
76             return []
77         node = self.root
78         for ch in pattern:
79             if ch not in node.children:
80                 return []
81             node = node.children[ch]
82         m = len(pattern)
83         return sorted(i for i in node.indices if i + m <= len(self.text) and self.text[i:
84
85
86
87 if __name__ == "__main__":
88     words = ["Fire", "Fira", "Firaga", "Firaja"]
89     trie = Trie(words)
90     tests = ["Fire", "Fira", "Firaga", "Firaja", "Firas", "Fi"]
91     print("Dictionary Trie tests:")
92     for w in tests:
93         print(f" {w!r}: {'FOUND' if trie.search(w) else 'NOT FOUND'}")
94
95     text = "anabanana"
96     st = SuffixTrie(text)
97     patterns = ["ana", "nana", "ban", "x", "a", "anabanana"]
98     print("\nSuffix Trie pattern searches in text:", text)
99     for p in patterns:
100         print(f" {p!r} -> indices {st.occurrences(p)}")
101
```

powershell ×

PS D:\1.SQM\1.UNIVERSIDAD\5. QUINTO SEMESTRE\2.Advanced Algorithms> & C:\Users\santy\AppData\Local\Microsoft\WindowsApps\python3.13.exe "d:/1.SQM/1.UNIVERSIDAD/5. QUIPS 0 SEMESTRE/2.Advanced Algorithms/Homework9_SuffixTrie/Hw9_SuffixTrie.py"
Dictionary Trie tests:
'Fire': FOUND
'Fira': FOUND
'Firaga': FOUND
'Firaja': FOUND
'Firas': NOT FOUND
'Fi': NOT FOUND

Suffix Trie pattern searches in text: anabanana
'ana' -> indices [0, 4, 6]
'nana' -> indices [5]
'ban' -> indices [3]
'x' -> indices []
'a' -> indices [0, 2, 4, 6, 8]
'anabanana' -> indices [0]
PS D:\1.SQM\1.UNIVERSIDAD\5. QUINTO SEMESTRE\2.Advanced Algorithms>

main ↺ ↻ ↺ ↻ ⏏ ⏏ 0 0

Go Live 🔔

<https://colab.research.google.com/drive/1c1l3gEze4-YopuSQPSfIcmdDvbpXY-cs?usp=sharing>

REFERENCES

GeeksforGeeks. (2025, August 8). *Pattern Searching using Suffix Tree*. GeeksforGeeks.

<https://www.geeksforgeeks.org/dsa/pattern-searching-using-suffix-tree/>

GeeksforGeeks. (2025, July 23). *Generalized Suffix tree*. GeeksforGeeks.

<https://www.geeksforgeeks.org/dsa/generalized-suffix-tree/>

Universidade do Porto, & Ribero, P. R. (2020). Competitive Programming - String Matching - Suffix Trie. *Competitive Programming - String Matching - Suffix Trie*, 1(1), <https://www.dcc.fc.up.pt/~pribeiro/aulas/pc1920/class07.html>.

<https://www.dcc.fc.up.pt/~pribeiro/aulas/pc1920/material/stringmatching.pdf>