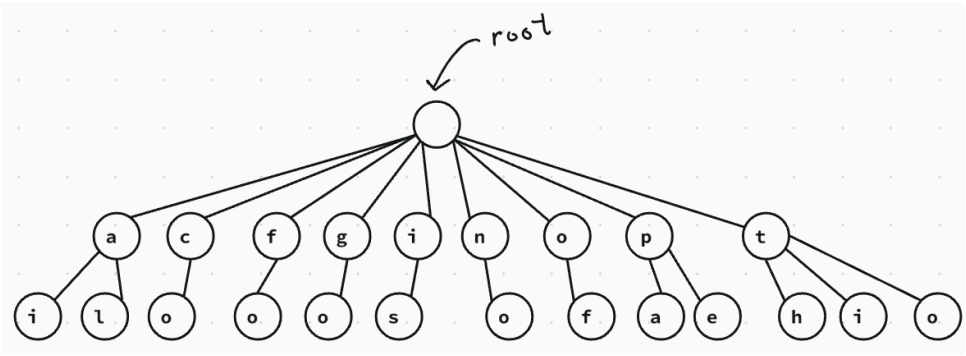

Figure 1: R-way trie for Exercise 1.

.

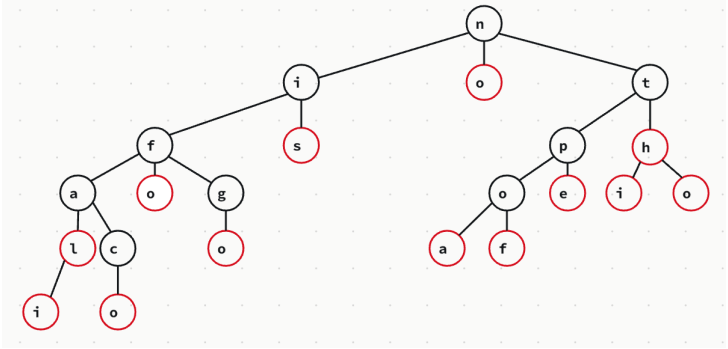

Figure 2: TST for Exercise 2.

.

Sergio E. Garcia Tapia
Algorithms by Sedgewick and Wayne (4th edition) [SW11]
January 25, 2025

5.2: Tries

Exercise 1. Draw the R-way trie that results when the keys

```
no is th ti fo al go pe to co to th ai of th pa
```

are inserted in that order into an initially empty trie (do not draw null links).

Solution. See Figure 1.

Exercise 2. Draw the TST that results when the keys

```
no is th ti fo al go pe to co to th ai of th pa
```

are inserted in that order into an initially empty TST.

Solution. See Figure 2.

Exercise 3. Draw the R-way trie that results when the keys

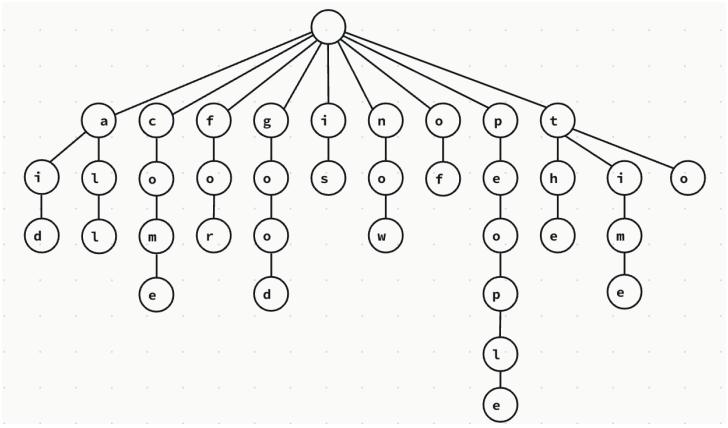

Figure 3: R-way trie for Exercise 3.

.

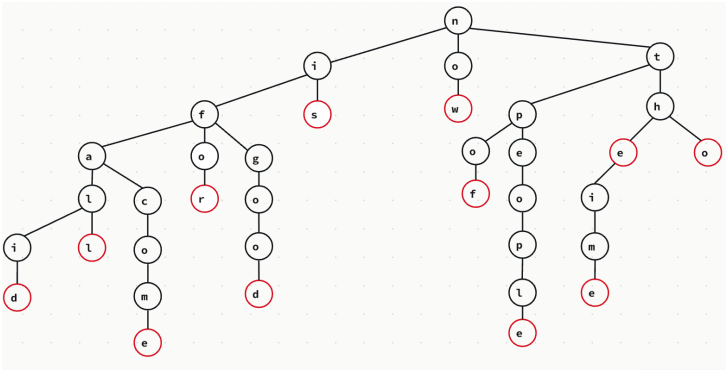

Figure 4: TST for Exercise 4.

.

```
now is the time for all good people to come to the aid of
```

are inserted in that order into an initially empty trie (do not draw null links).

Solution. See Figure 3.

Exercise 4. Draw the TST that results when the keys

```
now is the time for all good people to come to the aid of
```

are inserted in that order into an initially empty TST.

Solution. See Figure 4.

Exercise 5. Develop nonrecursive versions of `TrieST` and `TST`.

Solution. See `com.segarciat.algs4.ch5.sec2.ex05`.

Exercise 6. Implement the following API, for a `StringSET` data type:

```
      StringSET          // create a string set
void add(String key)     // put key into the set
```

```
   void delete(String key) // remove key from the set
boolean contains()         // is key in the set?
boolean isEmpty()       // is the set empty?
    int size()          // number of keys in the set
 String toString()      // string representation of the set
```

Solution. See `com.segarciat.algs4.ch5.sec2.ex06`.

Exercise 10. *Size*: Implement very eager `size()` (that keeps in each node the number of keys in its subtree) for `TrieST` and `TST`.

Solution. See `com.segarciat.algs4.ch5.sec.ex05`, my non-recursive implementations of these classes. There, I provided a very eager implementation of `size()`.

References

[SW11] Robert Sedgewick and Kevin Wayne. *Algorithms*. 4th ed. Addison-Wesley, 2011. ISBN: 9780321573513.