

Con Amore Special subjects

Jacob Trier Frederiksen and Anders Kalhauge



Spring 2019



Tries

Alphabets Strings

k-d trees

Big Data

What is big What about bugs



Tries

Alphabets Strings

k-d trees

Big Data

What is big What about bugs



An alphabet is

a finite set of characters

In our case it will also be

- ordered
- perfectly hashed

Alphabet examples



- \square English letters $\{A, B, \dots, Z\}$
- \square Decimal digits $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- \square Binary digits $\{0,1\}$
- ☐ Traffic light colours

Exercise 1 - Discuss in groups



What is meant by:

- ☐ Fininte set of characters
- Order of characters
- ☐ Perfect hashing of characters

Exercise 2 - Perfect hashing



In Java create functions, that can perfectly hash

- □ English letters, input is char
- ☐ digits, input is int
- red, yellow, green in that order, input is String

```
public int indexOf(char letter) {
   ...
}
```



A string is

a finite sequence of characters

from an alphabet

including the empty string $\boldsymbol{\epsilon}$



- \square Insertion: O(1)
- \square Search: O(1)
- \square Sort n elements: O(n)
- \square Range of m elements: O(m)



- \square Insertion: O(1)
- \square Search: O(1)
- \square Sort n elements: O(n)
- \square Range of m elements: O(m)

Like hashtables on steroids...



- \square Insertion: O(1)
- \square Search: O(1)
- \square Sort n elements: O(n)
- \square Range of m elements: O(m)

Like hashtables on steroids, but keys has to be alphabetic



- \square Insertion: O(1)
- \square Search: O(1)
- \square Sort n elements: O(n)
- \square Range of m elements: O(m)

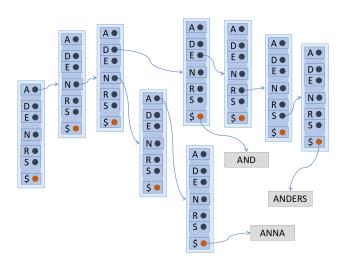
Like hashtables on steroids, but keys has to be alphabetic and the sorting is alphabetic too (lexicographic)



What are the properties of Hashtables compared to Tries

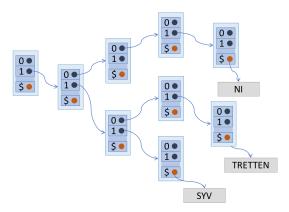
- \square Insertion: O(1)
- \square Search: O(1)
- \square Sort n elements: O(n)
- \square Range of m elements: O(m)





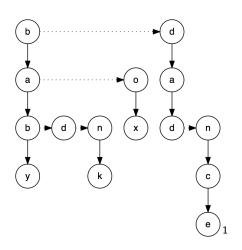
Trie with binary numbers





Trie implemented with linked lists





¹By Qwertyus - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=38468060

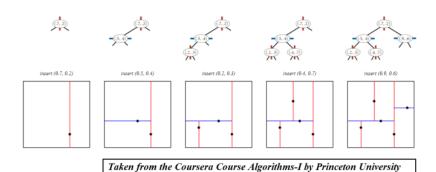


Complexities if the k-d tree is balanced

- \square Create: $O(n \log n \log n)$ using $O(n \log n)$ sorting
- \square Insert: $O(\log n)$
- \square Search: $O(\log n)$ nearest neighbour
- \square Range of m elements with k axis: $O(n^{1-\frac{1}{k}}+m)$
- \square Range of m elements with 2 axis: $O(\sqrt{n} + m)$

Building a 2-d tree dynamically





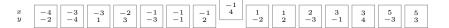


https://youtu.be/Z4dNLvno-EY

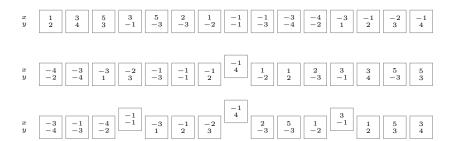




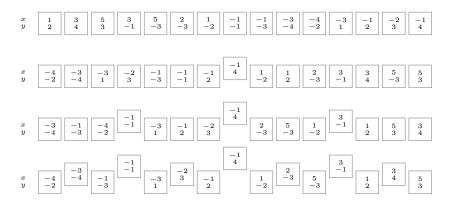




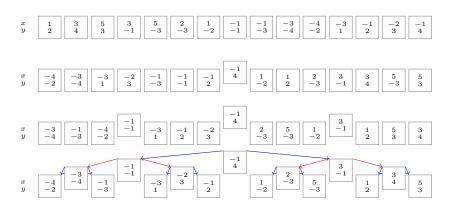




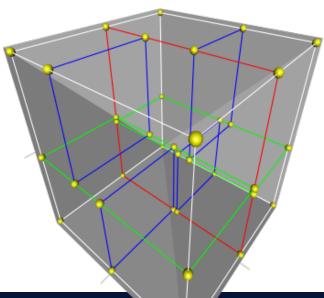














Tries

Alphabets Strings

k-d trees

Big Data

What is big What about bugs



- □ 1 Kb, kilobyte
 - □ 1.000 byte
- □ 1 Mb, megabyte
 - □ 1.000.000 byte
- □ 1 Gb, gigabyte
 - □ 1.000.000.000 byte
- □ 1 Tb, terrabyte
 - □ 1.000.000.000.000 byte
- □ 1 Pb, pettabyte
 - □ 1.000.000.000.000.000 byte



Extreme weather



Mapping Denmark...

 \square 43.000 km 2



- □ 43.000 km²
- \square average of 10 messure points per m²



- □ 43.000 km²
- \square average of 10 messure points per m²
- □ 1000 bytes of data per point



- □ 43.000 km²
- □ average of 10 messure points per m²
- □ 1000 bytes of data per point

1.000 * 10 * 1.000.000 * 43.000 = 430.000.000.000.000 = 430Tb



- □ 43.000 km²
- □ average of 10 messure points per m²
- □ 1000 bytes of data per point

$$1.000 * 10 * 1.000.000 * 43.000 = 430.000.000.000.000 = 430$$
Tb

How big is the chance of at least one disk error on 1.000 PC's?



