Data Structures and Algorithms

XMUT-COMP 103 - 2024 T1

Using Set

Mohammad Nekooei

School of Engineering and Computer Science

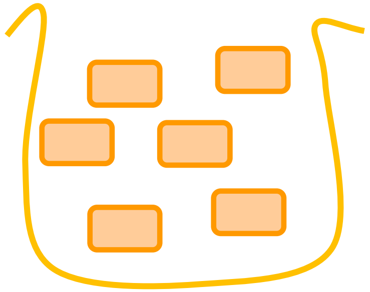
Victoria University of Wellington

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Using Sets

- Vocabulary:
  - Given a file of words (from a book)
  - Count the number of words and the number of distinct words.

    - open the file
    - initialise Vocab = new collection of String
    - initialise totalWordCount = 0
    - for each word in the file
      - increment totalWordCount
      - if the word is not in the vocab, then add it
    - return totalWordCount and the size of Vocab



This is the potentially expensive operation

  - What kind of Collection makes it efficient to check if the word is in the vocab already?

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

List

```
List<String> vocab = new ArrayList<String>();
try{
    Scanner sc = new Scanner(new File(filename));
    while (sc.hasNext()){
        String word = sc.next();
        if(!vocab.contains(word)) {
            vocab.add(word);
        }
    }
}
catch(IOException e){…}

UI.println("Number of different words: " + vocab.size());
```

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Set

```
Set<String> vocab = new HashSet<String>(); try{
    Scanner sc = new Scanner(new File(filename));
    while (sc.hasNext()){
        String word = sc.next();
        vocab.add(word);  //Notice no need to check vocab.contains(word) first
    }
}
catch(IOException e){…}

UI.println("Number of different words: " + vocab.size());

for(String s : vocab) {UI.println(s);} //Print each word
```

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

0	1	2	3	4	5	6	7	8	9

Set:

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I									
0	1	2	3	4	5	6	7	8	9

Set:

I									

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I									
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**								

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**								
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**								

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**								
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**							

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**								
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**							

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**							
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**							

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**							
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**						

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**							
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**						

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**							
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**						

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**						
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**						

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play						
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games					

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**						
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**					

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**						
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**					

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play						
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games					

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games					
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games					

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games					
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games					

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games					
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**					
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games					
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games					
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**					
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**	**also**				
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**	**also**				
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**	**also**				
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also				

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**	**also**				
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**	**make**			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also				
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	**like**	**to**	**play**	**games**	**also**				
0	1	2	3	4	5	6	7	8	9

Set:

I	**like**	**to**	**play**	**games**	**also**	**make**			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also	make			
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also	makes			
0	1	2	3	4	5	6	7	8	9

Set:

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also	makes			
0	1	2	3	4	5	6	7	8	9

Set: size() == 7 //DONE

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also	makes			
0	1	2	3	4	5	6	7	8	9

Set: size() == 7 //DONE

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also	makes			
0	1	2	3	4	5	6	7	8	9

Set: size() == 7 //DONE

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List:

I	like	to	play	games	also	makes			
0	1	2	3	4	5	6	7	8	9

Set: size() == 7 //DONE

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List: size() == 7 //DONE

I	like	to	play	games	also	makes			
0	1	2	3	4	5	6	7	8	9

Set: size() == 7 //DONE

I	like	to	play	games	also	make			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Example

TEXT: I like to play games. I also like to make games.

List: size() == 7 //DONE using 28 extra steps

I	**like**	**to**	**play**	**games**	**also**	**makes**			
0	1	2	3	4	5	6	7	8	9

Set: size() == 7 //DONE

I	**like**	**to**	**play**	**games**	**also**	**make**			

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

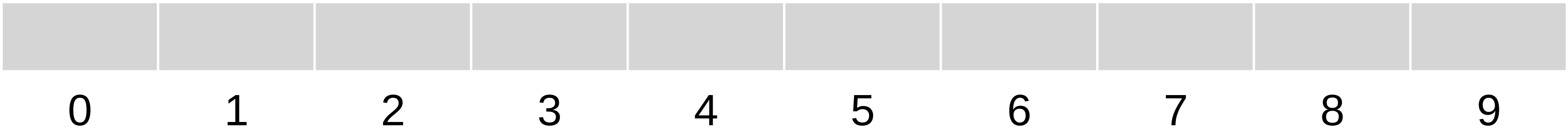
Sets: HashSet

- HashSets:
  - uses an array to store the values.
  - given a value, compute an index where it belongs (hashCode)
  - jump to that index in the array
  - **speed is independent of how big the set is!!!**

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

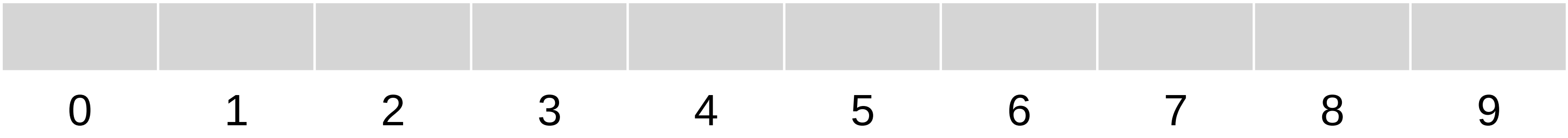
- Lets add the characters “a”, ”c”, ”q” and “a” to a HashSet



© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Lets add the characters “a”, ”c”, ”q” and “a” to a HashSet
- “a” hashCode => 97



© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Lets add the characters “a”, ”c”, ”q” and “a” to a HashSet
- “a” hashCode => 97 % 10 = 7

							a		
0	1	2	3	4	5	6	7	8	9

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Lets add the characters “a”, ”c”, ”q” and “a” to a HashSet
- “a” hashCode => 97 % 10 = 7
- “c” hashCode => 99 % 10 = 9

							a		**c**
0	1	2	3	4	5	6	7	8	9

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Lets add the characters “a”, ”c”, ”q” and “a” to a HashSet
- “a” hashCode => 97 % 10 = 7
- “c” hashCode => 99 % 10 = 9
- “q” hashCode => 113 % 10 = 3

			q				**a**		**c**
0	1	2	3	4	5	6	7	8	9

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Lets add the characters “a”, ”c”, ”q” and “a” to a HashSet
- “a” hashCode => 97 % 10 = 7
- “c” hashCode => 99 % 10 = 9
- “q” hashCode => 113 % 10 = 3
- “a” hashCode => 97

			q				**a**		**c**
0	1	2	3	4	5	6	7	8	9

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

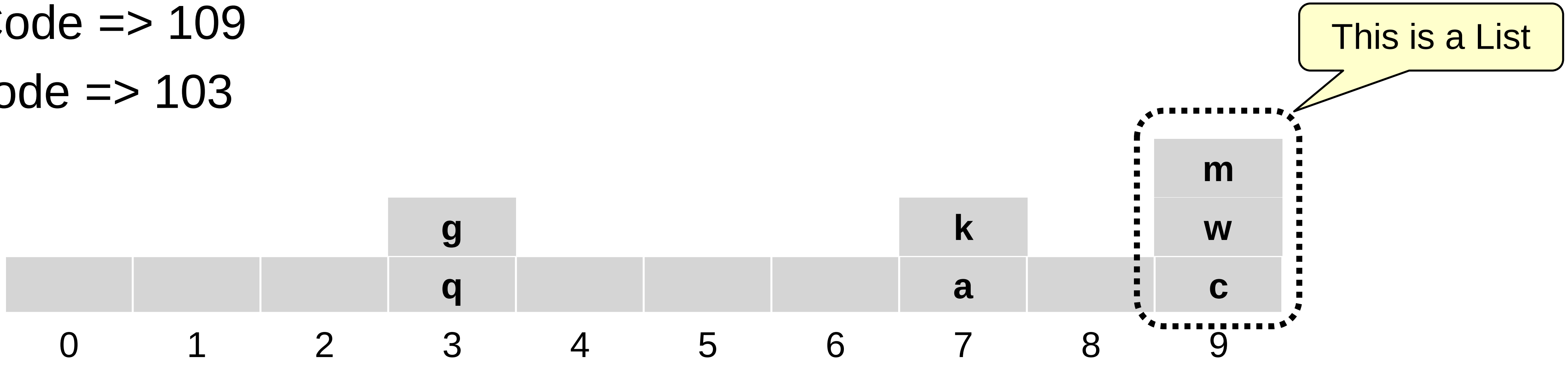
Sets: HashSet

- Problem: suppose two values have the same hashCode?
  - make a “bucket” – i.e. a list of values, and search down the list
  - OK, as long as the HashSet doesn’t get too full
  - If the HashSet gets a bit full (eg, 70%)
    - make a new array (double the size) and move all the values over

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Lets add more characters to a HashSet
- “a” hashCode => 97
- “c” hashCode => 99
- “q” hashCode => 113
- “k” hashCode => 107
- “w” hashCode => 119
- “m” hashCode => 109
- “g” hashCode => 103



© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Sets: HashSet

- Issue: Is the hashCode calculated correctly
  - Will each object have a unique code?
  - Are the values skewed/badly distributed?

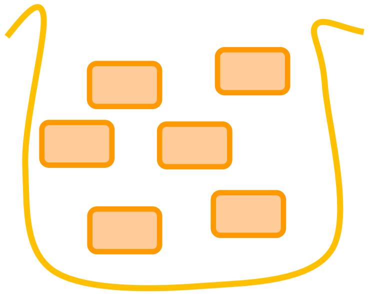
- Potential Problem: order of items is all mixed up

- Alternative method
  - Could we use the natural order of the elements to determine if they already are in the set?

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Recap: Using Sets

- Vocabulary:
  - Given a file of words (from a book)
  - Count the number of words and the number of distinct words.



    - open the file
    - initialise Vocab = new collection of String
    - initialise totalWordCount = 0
    - for each word in the file
      - increment totalWordCount
      - if the word is not in the vocab, then add it
    - return totalWordCount and the size of Vocab

This is the potentially expensive operation

  - What kind of Collection makes it efficient to check if the word is in the vocab already?

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Using Sets: Vocabulary, again

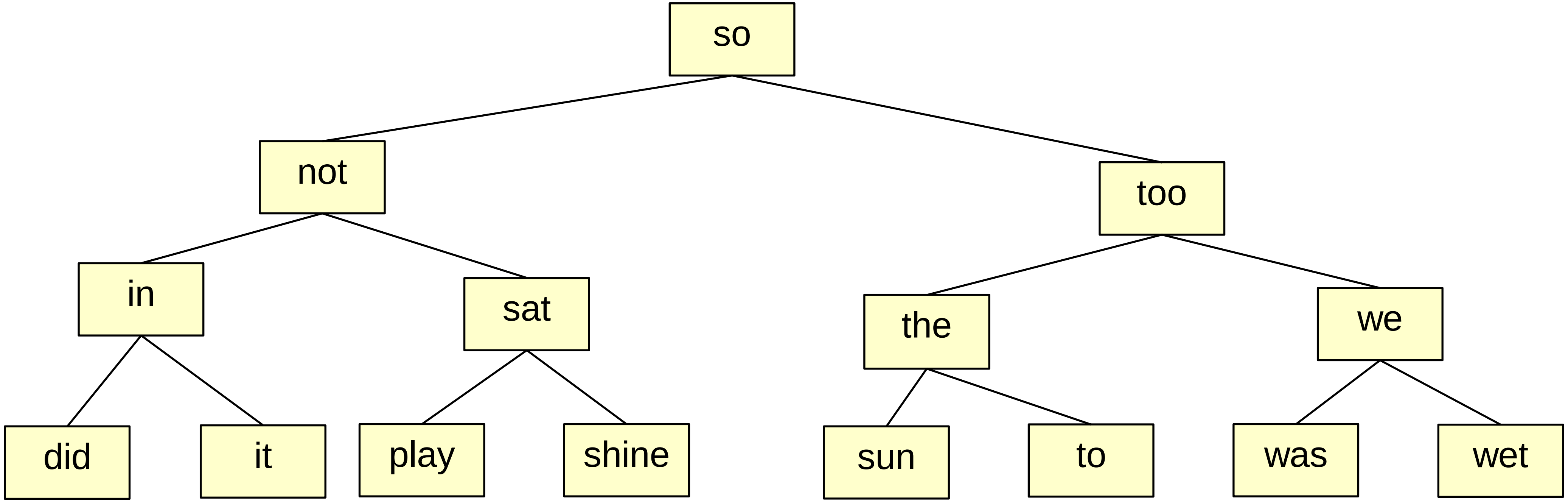
- Vocabulary:
  - Given a file of words (from a book)
  - Count the number of words and the number of distinct words.
  - Print out the vocabulary:
    - all words, alphabetically
  - How can we sort the words?

```
List<String> sortedVocab = new ArrayList<String>(Vocab);
                         // or create empty then add all:  sortedVocab.addAll(vocab);
Collections.sort(sortedVocab);
for (String word : sortedVocab){
    outfile.println(word);
}
```

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

TreeSet

- TreeSet: a class that implements Set (and SortedSet)
  - Keep all the values in a tree structure, alphabetically organised.
  - Search down the branches to find values



  - Not quite as fast as HashSets, but very close!
  - Million items – about 20 steps maximum to find any item.
  - Around 20,000,000 steps to add 1,000,000 items.

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Using TreeSet: Vocabulary, again

```
Set<String> sortedVocab = new TreeSet<String>();

while (scan.hasNext()){
    sortedVocab.add(scan.next());
}
for (String word : sortedVocab){
    outfile.println(word);
}
```

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei

Measuring the performance

- Run the VocabularyMeasurer program
  - Counts vocabulary of a file using HashSet, TreeSet, and ArrayList.
  - Measures and reports the time taken.

  - Key question: Does it matter which one we use?

© Peter Andreae, Karsten Lundqvist, and Mohammad Nekooei