Programming for Mobile Computing

EECS 1022

`moodle.yorku.ca`

Anomalous grade distribution

If more than 30% of the students receive an A+ or A, then the grade distribution is deemed anomalous. As a result, expect it to be more difficult to receive an A+ or A in the programming part of the final exam.

Mistake in the marking of your midterm

If you believe that there is a mistake made in the marking of your midterm (the marking scheme is included in the feedback that has been emailed to you), then email the instructor within one week (that is, before Monday July 31). In the email, clearly describe the mistake in marking. Your whole midterm will be reviewed. As a result your mark may go up, stay the same, or go down.

Collection

- Words in a dictionary.
- Foot print of a word.
- Students in a class.
- Grades of a student.
- Songs in an iTunes library.
- ...

Collection

Three types of collection that are used often:

- List
- Set
- Map

List versus Set

A list may contain duplicates whereas a set does not contain any duplicates.

List versus Set

A list may contain duplicates whereas a set does not contain any duplicates.

- If we add the element 3 to the end of the list $[1, 3, 2]$ then we obtain the list $[1, 3, 2, 3]$.
- If we add the element 3 to the set $\{1, 3, 2\}$ then the set does not change and remains $\{1, 3, 2\}$.

List versus Set

The elements of a list are ordered whereas the elements of a set are not ordered.

List versus Set

The elements of a list are ordered whereas the elements of a set are not ordered.

- The lists $[1, 3, 2]$ and $[3, 2, 1]$ are different lists.
- $\{1, 3, 2\}$ and $\{3, 2, 1\}$ represent the same set.

List versus Set

Question

How would you represent a collection of grades of a student (so that you can compute the student's GPA)? List or Set? Motivate your answer.

List versus Set

Question

How would you represent a collection of grades of a student (so that you can compute the student's GPA)? List or Set? Motivate your answer.

Answer

List. A student may have multiple A+s. To compute the student's GPA we need those duplicates.

List versus Set

Question

How would you represent a collection of student IDs (so that you can keep track who has completed a lab)? List or Set? Motivate your answer.

List versus Set

Question

How would you represent a collection of student IDs (so that you can keep track who has completed a lab)? List or Set? Motivate your answer.

Answer

Set. There is no need for ordering or duplicates.

List versus Set

Question

How would you represent a collection of words (so that you can sort them for efficient look up)? List or Set? Motivate your answer.

List versus Set

Question

How would you represent a collection of words (so that you can sort them for efficient look up)? List or Set? Motivate your answer.

Answer

List. For sorting, an ordering is essential.

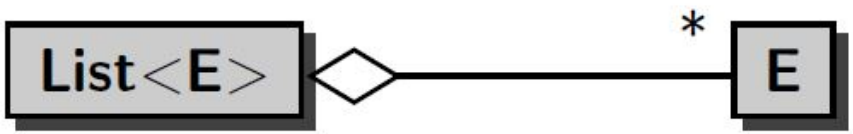
Lists

Lists can be classified based on

- the type of the elements of the list (Integer, Double, Boolean, ...) and
- the way the list is implemented (using an array, using "links," ...).

Lists

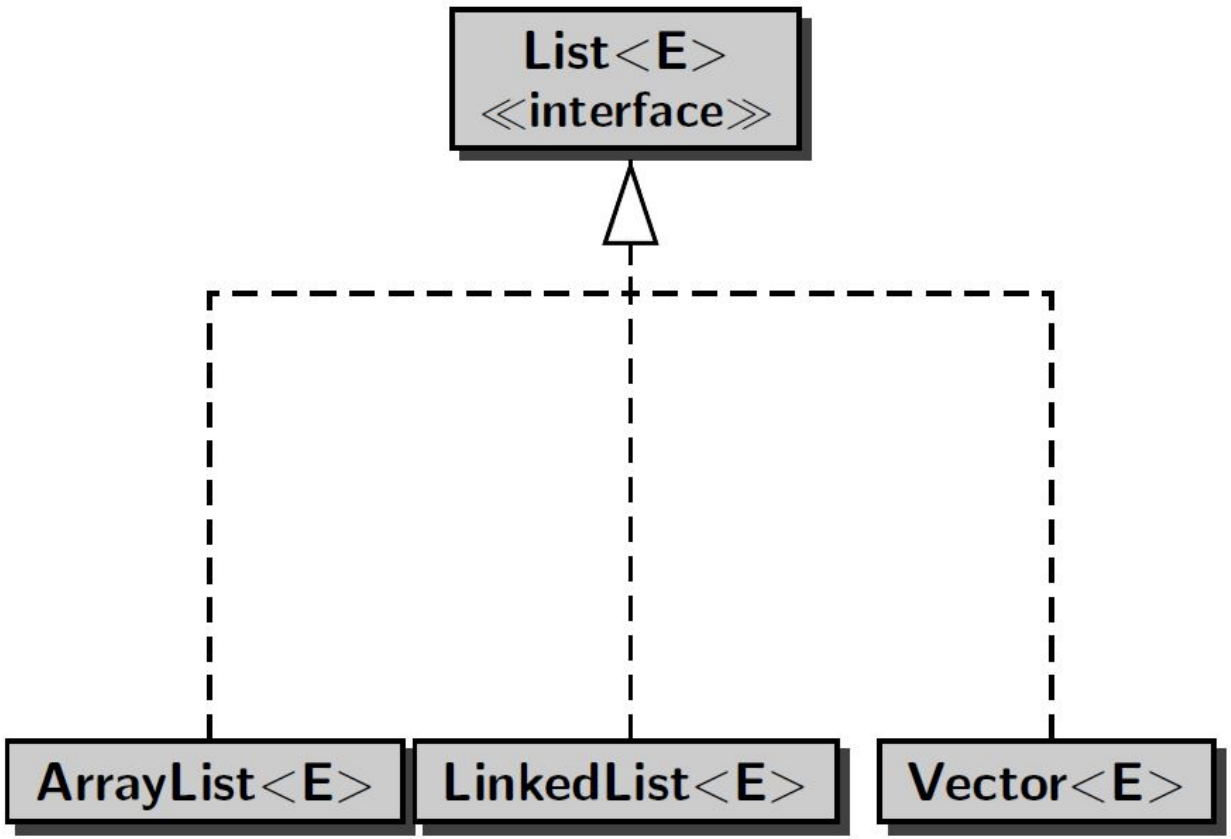
To abstract from the type of the elements of the list, we exploit generics.



E is a type parameter. The elements of the list are of type E.

Lists

To abstract from the way the list is implemented, we exploit interfaces.



java.util

Interface List<E>

Type Parameters:

E - the type of elements in this list

All SuperInterfaces:

Collection<E>, Iterable<E>

All Known Implementing Classes:

AbstractList, AbstractSequentialList, ArrayList,

ArrayList

java.util

Class ArrayList<E>

java.lang.Object
    java.util.AbstractCollection<E>
        java.util.AbstractList<E>
            java.util.ArrayList<E>

All Implemented Interfaces:

`Serializable, Cloneable, Iterable<E>, Collection<E>, List<E>,`

Class versus Interface

interface	specification	what?
class	implementation	how?

Collection of words

```
List<String> dictionary = new ArrayList<String>();
```

- The type of the elements is `Integer` and
- the list is implemented by means of an array.

Collection of words

```
List<String> dictionary = new ArrayList<String>();
```

- The type of the elements is `Integer` and
- the list is implemented by means of an array.

Question

Why can we assign an object of type `ArrayList<String>` to a variable of type `List<String>`?

Collection of words

```
List<String> dictionary = new ArrayList<String>();
```

- The type of the elements is `Integer` and
- the list is implemented by means of an array.

Question

Why can we assign an object of type `ArrayList<String>` to a variable of type `List<String>`?

Answer

Because the class `ArrayList<E>` implements the interface `List<E>`.

Grades of a student

```
List<Double> grades = new LinkedList<Double>();
```

- The type of the elements is `Double` and
- the list is implemented by means of "links."

ArrayList, LinkedList or Vector?

Depends on which operations on the list are performed.

Question

How many milliseconds does it take to add n elements to the end of a list?

ArrayList, LinkedList or Vector?

Depends on which operations on the list are performed.

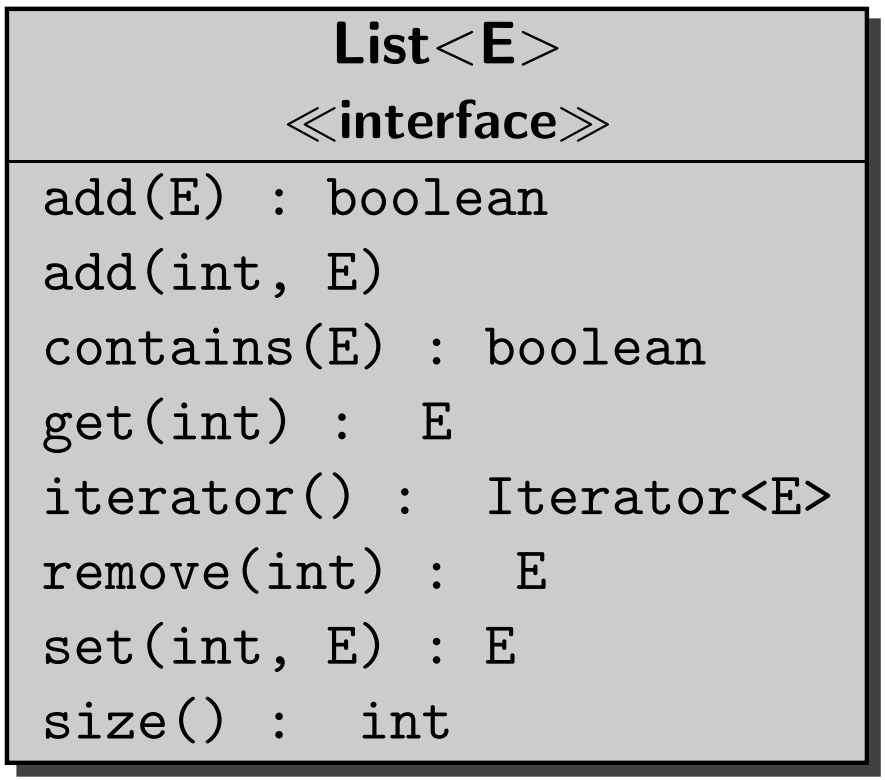
Question

How many milliseconds does it take to add n elements to the end of a list?

Answer

n	`ArrayList`	`LinkedList`	`Vector`
10^5	9	12	14
10^6	47	92	113
10^7	442	824	1041
2×10^7	913	1,650	2,076
3×10^7	1,350	143,616	3,230
4×10^7	2,527		4,103
5×10^7	2,689		6,119

Methods of List



Detecting duplicates

Problem

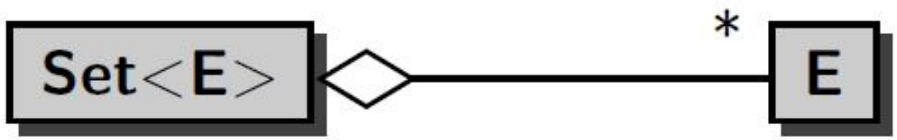
The ePost files are of the following format.

```
shuimt          5
demon202        5
maliko          5
bagcilar        5
yiyao           5
f4rdeen         5
keddy123        5
zizheng         5
```

Twice a week, several teaching assistants provide input for these files. This leads to duplication every now and then. How do we detect duplication?

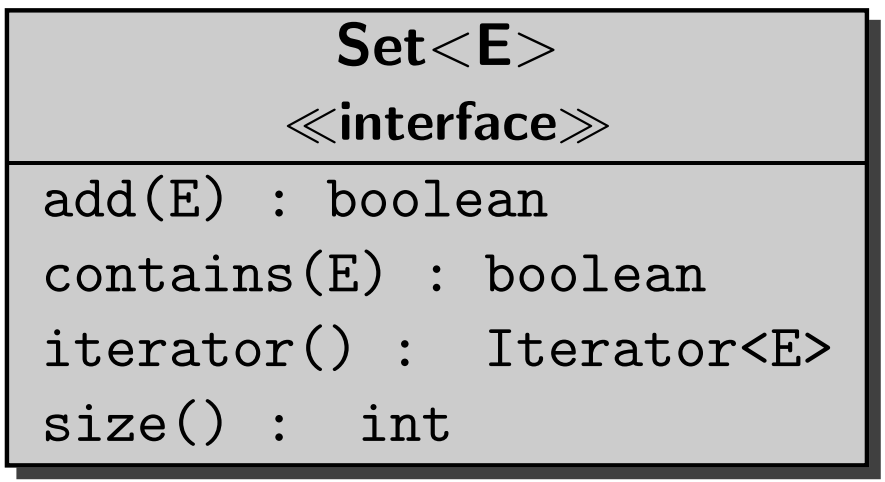
Sets

To abstract from the type of the elements of the set, we exploit generics.

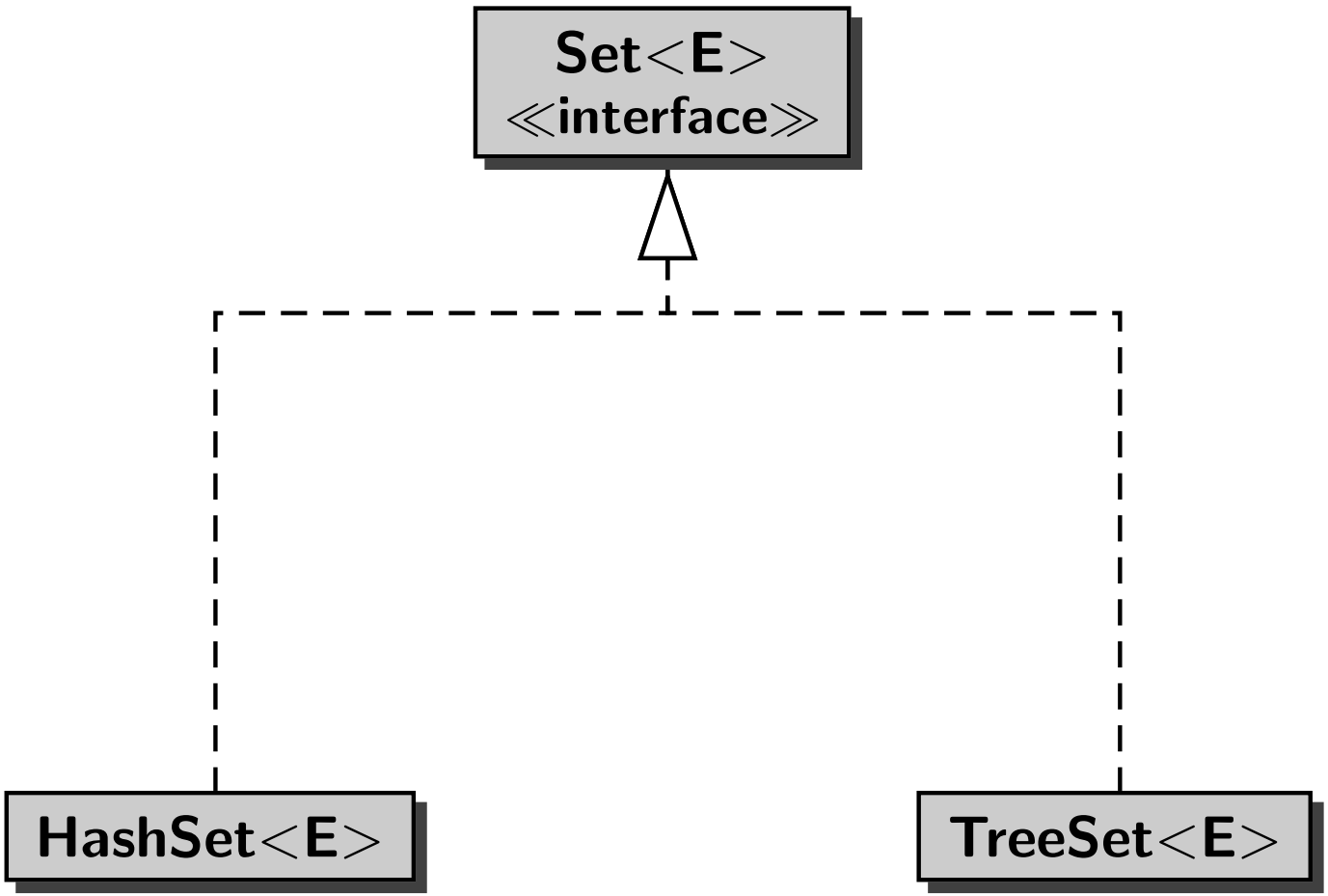


E is a type parameter. The elements of the set are of type E.

Methods of Set



Sets



Detecting duplicates

Problem

The ePost files are of the following format.

```
shuimt          5
demon202        5
maliko          5
bagcilar        5
yiyao           5
f4rdeen         5
keddy123        5
zizheng         5
```

Twice a week, several teaching assistants provide input for these files. This leads to duplication every now and then. How do we detect duplication?

Lists: indexed access

Each element of a list has an index.

```
List<String> dictionary = ...;
String word = dictionary.get(4);
```

```
  the
  of
  and
  to
  a
  in
  for
...
```

Lists: indexed access

Each element of a list has an index.

```
List<String> dictionary = ...;
String word = dictionary.get(4);
```

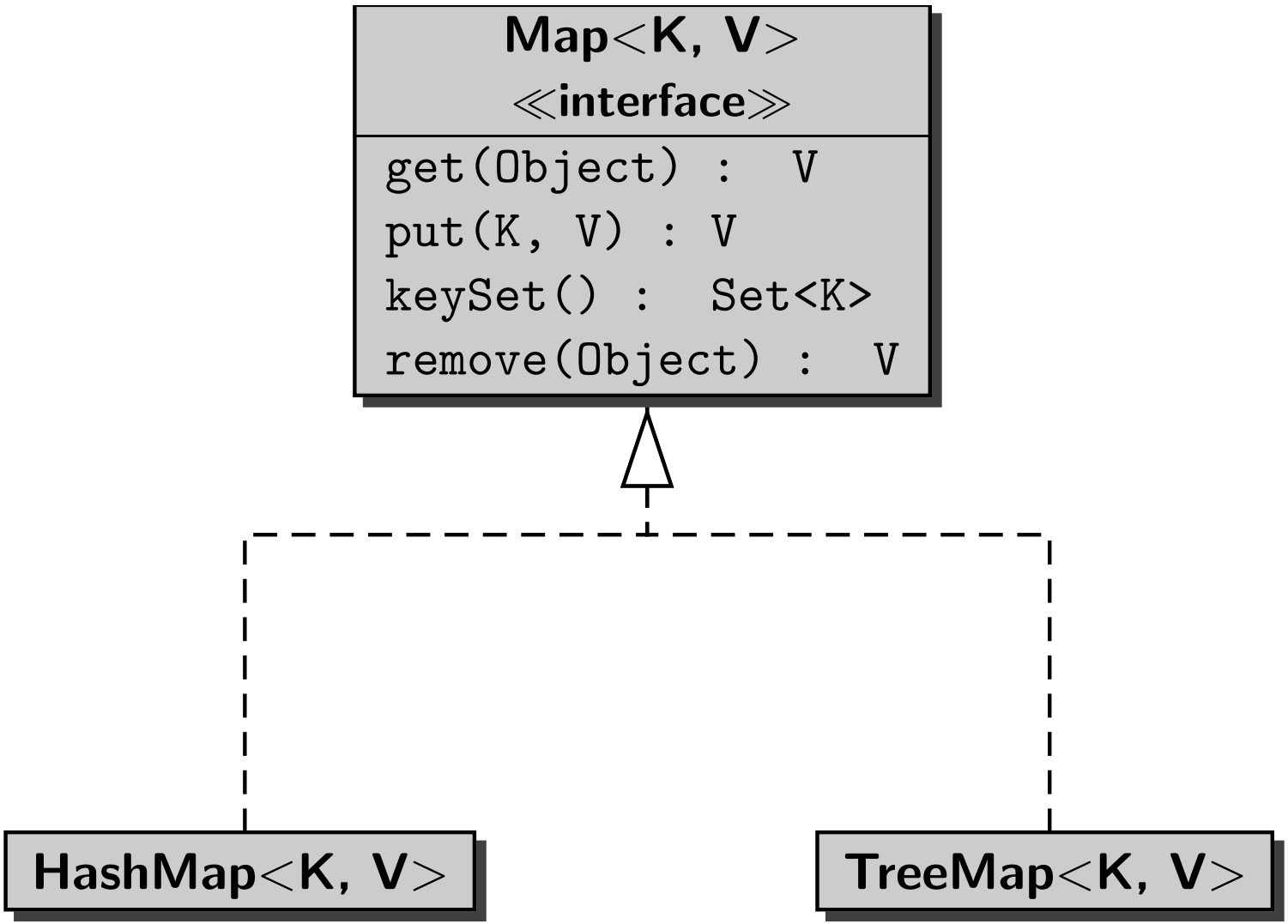
```
0 the
1 of
2 and
3 to
4 a
5 in
6 for
...
```

Indexed access

```
shuimt          5
demon202        5
maliko          5
bagcilar        5
yiyao           5
f4rdeen         5
keddy123        5
zizheng         5
```

Here, the indices (also known as keys) are Strings and the values are Integers.

Map


Map<K, V>
≪interface≫
get(Object) : V
put(K, V) : V
keySet() : Set<K>
remove(Object) : V
HashMap<K, V>
TreeMap<K, V>

Detecting duplicates

Problem

The ePost files are of the following format.

```
shuimt          5
demon202        5
maliko          5
bagcilar        5
yiyao           5
f4rdeen         5
keddy123        5
zizheng         5
```

Twice a week, several teaching assistants provide input for these files. This leads to duplication every now and then. How do we detect duplication?