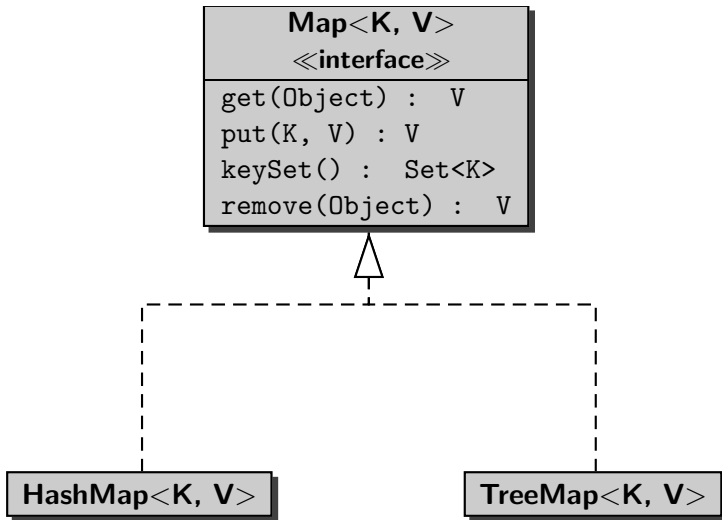


Map



The temperature app

- randomly selects a city in Ontario,
- reads a corresponding URL, and
- extracts the current temperature.

For each city, we need a corresponding URL. These can be stored in a file.

```
on-122_metric_e.html  Alexandria
on-1_metric_e.html   Algonquin Park (Brent)
on-29_metric_e.html  Algonquin Park (Lake of Two Rivers)
on-114_metric_e.html Alliston
on-30_metric_e.html  Apsley
on-111_metric_e.html Armstrong
on-148_metric_e.html Atikokan
on-164_metric_e.html Attawapiskat
...
```

Question

What is the most appropriate collection to store the cities and their URLs? A list, a set or a map?

Question

What is the most appropriate collection to store the cities and their URLs? A list, a set or a map?

Answer

A map.

Question

What is the most appropriate collection to store the cities and their URLs? A list, a set or a map?

Answer

A map.

Question

What are the types of the keys and values of the map?

Question

What is the most appropriate collection to store the cities and their URLs? A list, a set or a map?

Answer

A map.

Question

What are the types of the keys and values of the map?

Answer

String and URL.

Problem

Given a file containing cities and their URLs, create a Map.

Object Serialization

Rather than reading a string representation of an object from a file and creating the object, we can also read the object from a file directly.

```
ObjectInputStream objectInput =  
    new ObjectInputStream(  
        new FileInputStream("cities.ser"));  
Object object = objectInput.readObject();  
if (object instanceof Map)  
{  
    Map<String, URL> map = (Map) object;  
}  
objectInput.close();
```

Object Serialization

Rather than writing a string representation of an object to a file, we can also save the object to a file directly.

```
ObjectOutputStream objectOutput =  
    new ObjectOutputStream(  
        new FileOutputStream("cities.ser"));  
objectOutput.writeObject(map);  
objectOutput.close();
```

Question

Which objects can be serialized?

Object Serialization

Question

Which objects can be serialized?

Answer

Those objects that are an instance of a class that implements the interface `Serializable`.

Problem

Check whether a given word appears in the book entitled “The Java Language Specification, Java SE 8 Edition.” The book is contained in the file `jls8.pdf`.

Question

Consider

```
List<String> words = ...;  
String search = ...;  
boolean found = false;  
for (String word : words)  
{  
    found = word.equals(search) || found;  
}
```

Given that the list contains n elements, how many times is the method `equals` invoked?

Question

Consider

```
List<String> words = ...;  
String search = ...;  
boolean found = false;  
for (String word : words)  
{  
    found = word.equals(search) || found;  
}
```

Given that the list contains n elements, how many times is the method `equals` invoked?

Answer

n times.

```
int index = Collections.binarySearch(list, element);
```

- Precondition: the list is sorted.
- If the element is contained in the list then the method returns the index at which the element can be found.
- If the element is not in the list then the method returns a negative number.

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 11;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

index gets assigned the value 4.

Binary search

```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

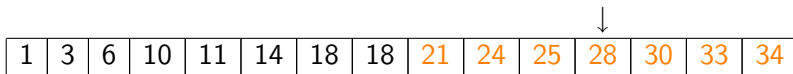
```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```

↓

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

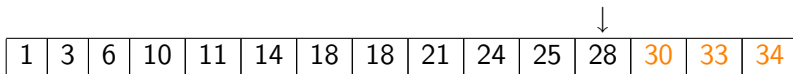
```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```



1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```



1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Binary search

```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

↓

Binary search

```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

↓

Binary search

```
final int ELEMENT = 32;  
int index = Collections.binarySearch(list, ELEMENT);
```

1	3	6	10	11	14	18	18	21	24	25	28	30	33	34
---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

index gets assigned the value -12 .


```
public static <T> int binarySearch(  
    List<? extends Comparable<? super T>> list,  
    T key)
```

This is a generic method. T is a type parameter.

Question

Which types match ? `extends Comparable< super T>?`

Question

Which types match ? `extends Comparable< ? super T>?`

Answer

Those types R such that

- R implements the interface `Comparable<S>` for some type S such that
- $S = T$ or S is a superclass of T.

Those types R such that

- R implements the interface `Comparable<S>` for some type S such that
- $S = T$ or S is a superclass of T .

Question

Take R , S and T to be `String`. Does the above hold?

Those types R such that

- R implements the interface `Comparable<S>` for some type S such that
- $S = T$ or S is a superclass of T .

Question

Take R , S and T to be `String`. Does the above hold?

Answer

Yes.

System.nanoTime()

This method returns the current value of the running Java Virtual Machine's high-resolution time source, in nanoseconds.

```
long start = System.nanoTime();  
...  
long stop = System.nanoTime();  
// stop - start is an estimate of the number of  
// nanoseconds it took to execute ...
```

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Binary search

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Answer

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c								

Binary search

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Answer

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1							

Binary search

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Answer

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2						

Binary search

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Answer

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2					

Binary search

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Answer

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3				

Binary search

Question

Consider

```
int index = Collections.binarySearch(list, ELEMENT);
```

Given that the list contains n elements, in the worst case, how many comparisons does the invocation of `binarySearch` make?

Answer

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4
2^{c-1}								

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4
2^{c-1}	1	2	2	4	4	4	4	8

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4
2^{c-1}	1	2	2	4	4	4	4	8

$$2^{c-1} \leq n$$

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4
2^{c-1}	1	2	2	4	4	4	4	8

$$2^{c-1} \leq n$$

$$\log_2(2^{c-1}) \leq \log_2(n)$$

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4
2^{c-1}	1	2	2	4	4	4	4	8

$$2^{c-1} \leq n$$

$$\log_2(2^{c-1}) \leq \log_2(n)$$

$$c - 1 \leq \log_2(n)$$

Answer continued

Let n be the number of elements of the list and c the number of comparisons needed in the worst case.

n	1	2	3	4	5	6	7	8
c	1	2	2	3	3	3	3	4
2^{c-1}	1	2	2	4	4	4	4	8

$$2^{c-1} \leq n$$

$$\log_2(2^{c-1}) \leq \log_2(n)$$

$$c - 1 \leq \log_2(n)$$

$$c \leq \log_2(n) + 1$$

Fact

Sorting a list of n elements needs approximately $n \log(n)$ comparisons.