

Binary Search Tree in Java

Prof. Michele Loreti

Programmazione Avanzata

Corso di Laurea in Informatica (L31)

Scuola di Scienze e Tecnologie

Binary Search Tree in Java



We want to develop an API for *immutable Binary Search Tree* in Java.

Binary Search Tree in Java

We want to develop an API for *immutable Binary Search Tree* in Java.

Our API consists of . . .

Binary Search Tree in Java

We want to develop an API for *immutable Binary Search Tree* in Java.

Our API consists of. . .

1. An interface `BSTree` that contains all the functionalities a *Binary Search Tree* must provide;

Binary Search Tree in Java

We want to develop an API for *immutable Binary Search Tree* in Java.

Our API consists of. . .

1. An interface `BSTree` that contains all the functionalities a *Binary Search Tree* must provide;
2. Classes that implement interface `BSTree`:

Binary Search Tree in Java

We want to develop an API for *immutable Binary Search Tree* in Java.

Our API consists of . . .

1. An interface `BSTree` that contains all the functionalities a *Binary Search Tree* must provide;
2. Classes that implement interface `BSTree`:
 - `BSTreeEmpty`, that represents an empty tree;

Binary Search Tree in Java

We want to develop an API for *immutable Binary Search Tree* in Java.

Our API consists of . . .

1. An interface `BSTree` that contains all the functionalities a *Binary Search Tree* must provide;
2. Classes that implement interface `BSTree`:
 - `BSTreeEmpty`, that represents an empty tree;
 - `BSTreeNode`, that represents a tree with at least one element.

Interface BSTree. . .

Interface BSTree is parametrised with the type T of handled elements:

```
public interface BSTree<T> {  
  
}
```


Interface BSTree. . .

Interface BSTree is parametrised with the type T of handled elements:

```
public interface BSTree<T> {  
  
}
```

Which methods you want to add to this interface?

Interface BSTree. . .

Interface BSTree is parametrised with the type T of handled elements:

```
public interface BSTree<T> {  
  
}
```

Which methods you want to add to this interface?

- T add(T value)
- boolean find(T v)
- int size ()
- boolean isEmpty()
- BSTree<T> remove(T value)

Implementing BSTree. . .

Class BSTreeEmpty describes an *empty* tree.

Class BSTreeNode describes a tree with at least one element.

Utility methods can be added to interface BSTree<T> to implement recurrent operations and simplify the use of the API.

To be continued...