

CS542 Project Two

Xiaoyan Sun & Shi Wang

1. Design

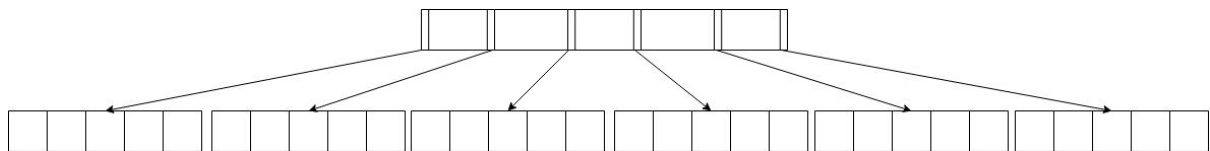
1.1 Assumptions

- The keys are distinct in B+tree.
- One specific key can have different values in store.
- The max entry number of the leaf is 5.

2. Algorithm

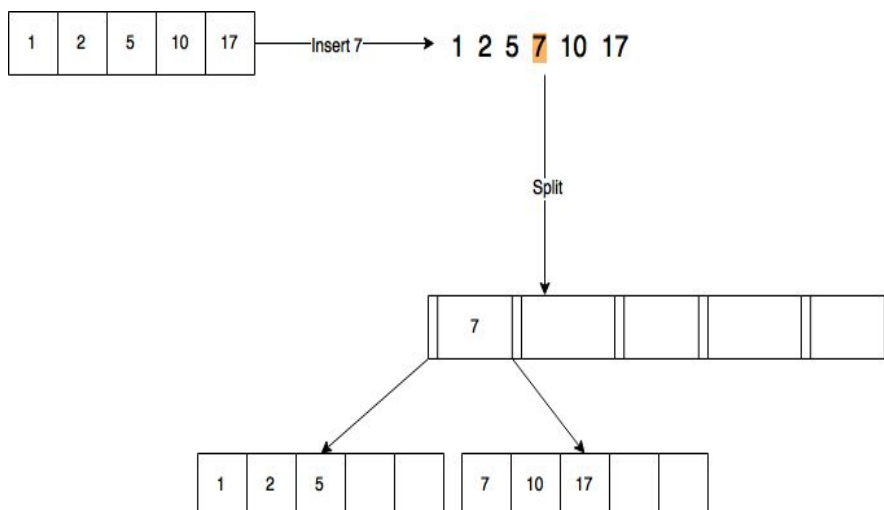
2.1 B+tree Model

- We use B+tree as the data structure dealing with the movie list searching.



2.2 B+tree Split

- When the leafnodes are full(number of leafnodes is 5), we should split the leaf while the next key inserting and put the median into the innernodes.
- After adding the next insert, we take the forth smallest number as the median(the smallest in the right half).
- Then we split the leaf into two halves and create the new innernode. Then we set the left half of the leaf as the innernode's left child and the right half as the right child.



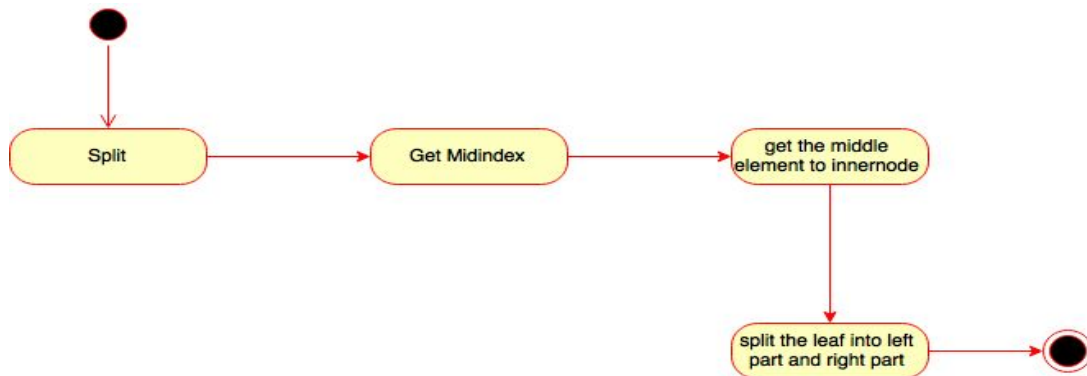
abstract of code:

```
//method
public bTreeNode<TKey> split(TKey key, int midIndex)
//First, create a new node
```

```

leafNode<TKey, TValue> newRNode = new leafNode<TKey, TValue>();
//Second, set the old key and value into new node
newRNode.setKey(i - midIndex, this.getKey(i));
newRNode.values[i-midIndex]=this.values[i];
//Insert Parent into the innernode
innerNode<TKey> theParent=(innerNode<TKey>) this.getParent();
newRNode.setParent(theParent);
theParent.insertAt(theParent.search(key),key,this,newRNode);
//return the new innernode
return theParent;

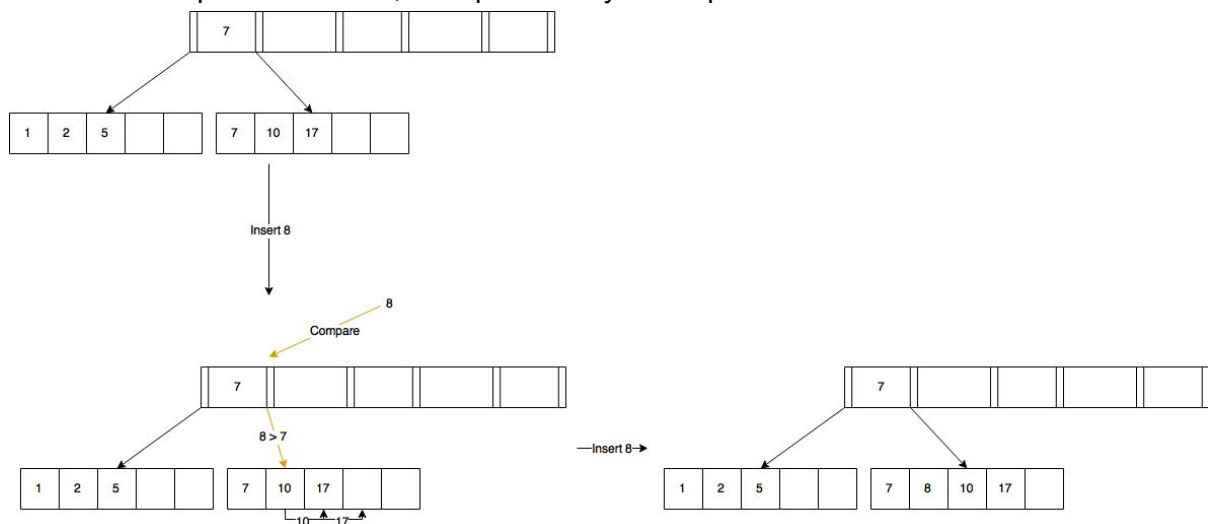
```



2.3 B+tree Put

void Put(string key, Number data_value); or void Put(string key, string data_value); adds the index entry.

- When we insert a new element into the B+tree, we should find the position for inserting and empty the position to insert the element.
- First, we find the position to insert. When a new key was inserted, the key should compare with the innernodes from the root to the leafnodes at the leaves and get the position between a smaller leafnode and a bigger leafnode.
- Second, we empty the target position by moving the bigger leafnodes one position behind, then put the key at the position.



abstract of code:

//First, run put function

public void Put(TKey key, TValue value)

//Second, goto insert function

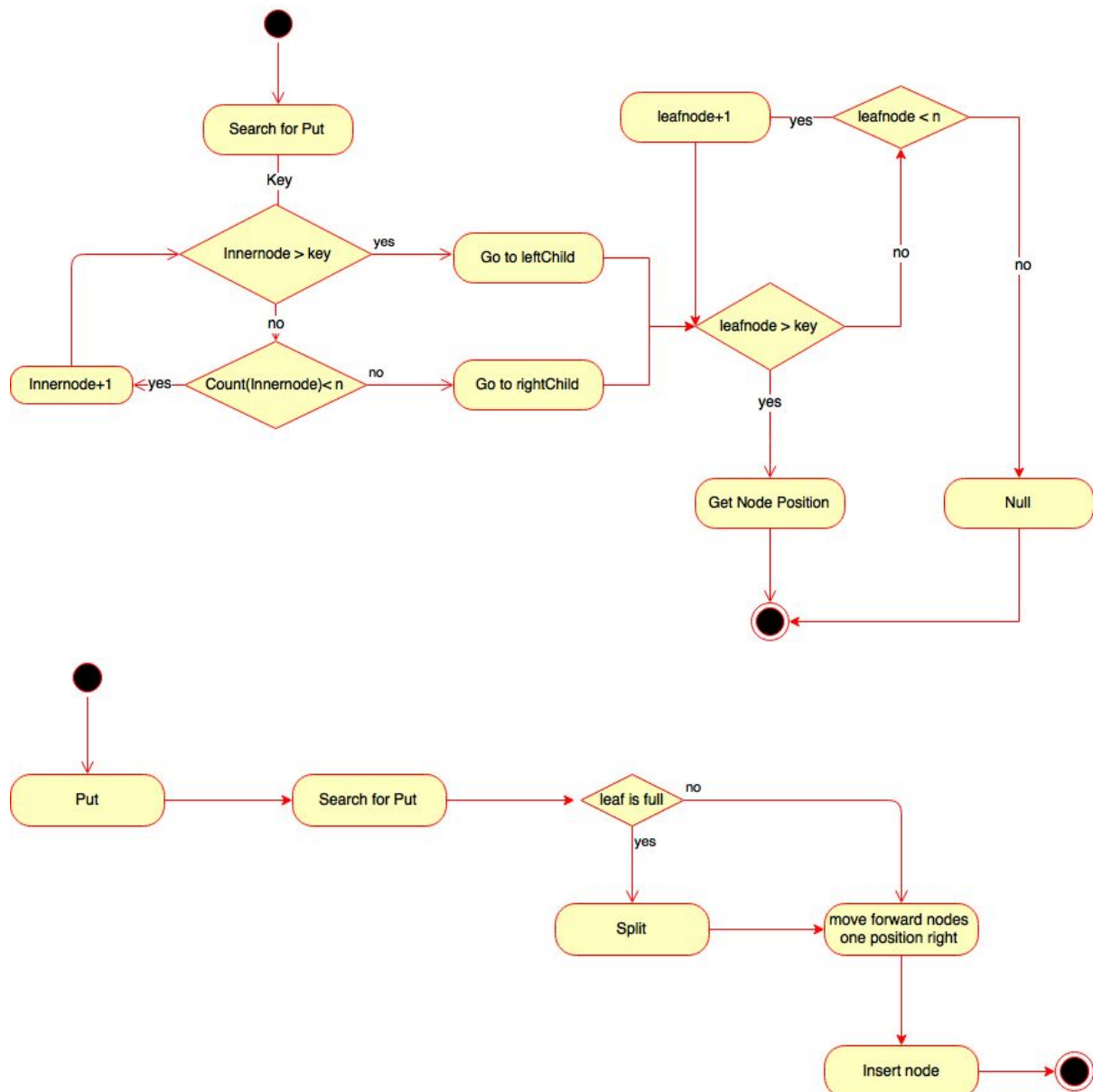
public void insert(TKey key)

//Third, goto insertKey function

public void insertKey(TKey key, TValue value)

//Forth, goto insertAt function to get the position to put. Then set the key and value.

private void insertAt(int index, TKey key, TValue value)

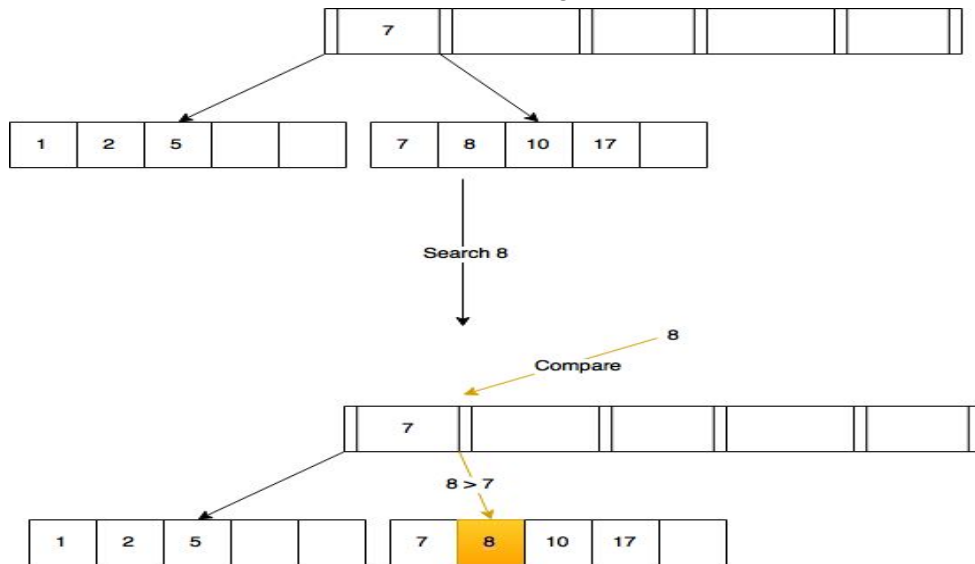


2.4 B+tree Get

string Get(Number data_value); or string Get(string data_value); retrieves the key given the index and

- When we are searching the specific element in the B+tree, we should find the position of the element and print out the value.

- First, we compare the element with the innernodes start from root. If bigger than the innernode, go to the right child. If smaller, go to the left child.
- Second, we search the element in the leafnodes. If the element equles the leafnode, then print out the searching value.



abstract of code:

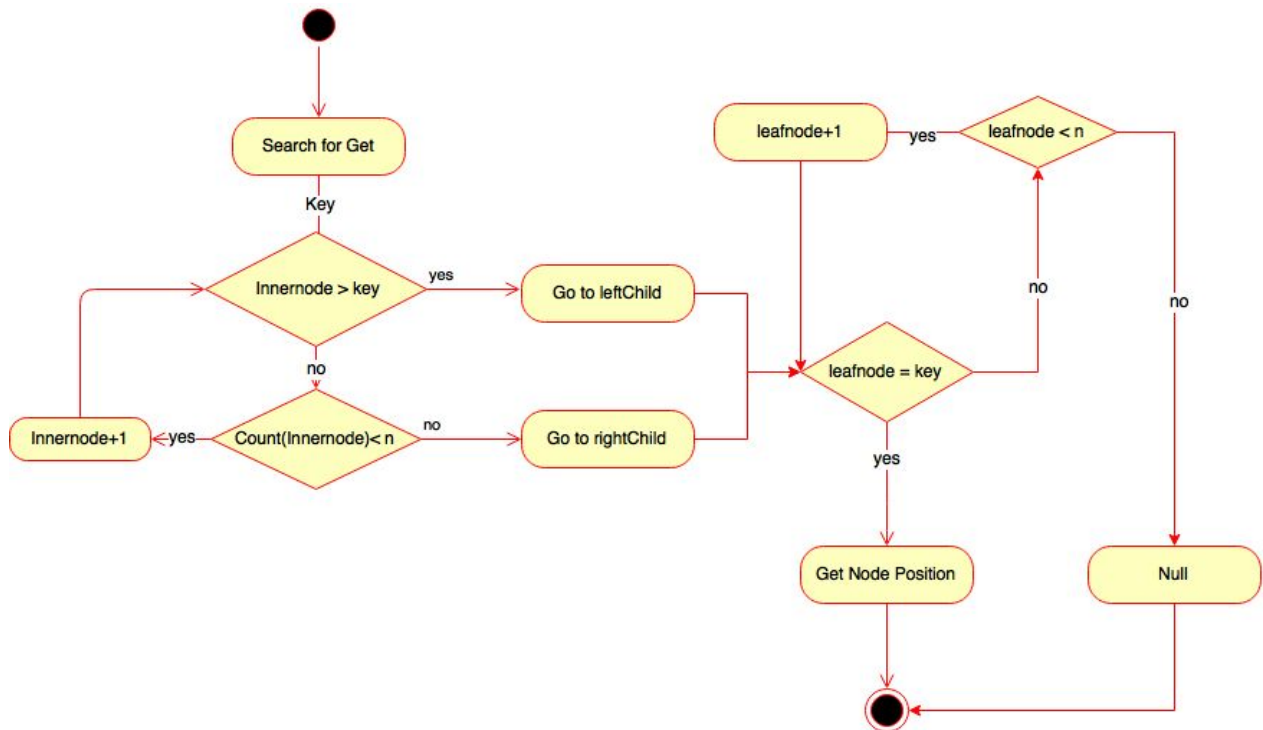
//First, run get function

public void Get(TKey key)

//Second, goto search function to get the position of the key.

public int search(TKey key)

//Third, printout the key and value





2.5 B+tree Remove

void Remove(string key); deletes the index.



3. test and result

3.1 The index in this case is defined as *CREATE INDEX movieInd ON Movie(year, format)*. The data value could be a composite string. For the movie *Ghost* for example, it could be "1990|laserdisk". Using the index you created,

- 3.1.1 Find all DVD movies made in 1977.
- 3.1.2 Find all VHS movies made in 1990.
- 3.1.3 Find all DVD movies made in 2001.

(1)method and process

- We insert the key and value by reading the file with split function, after insert all the elements, we search the DVD movies made in 1997, VHS movies made in 1990 and DVD movies made in 2001.

(2)result

```

Problems Javadoc Declaration Console
<terminated> test2 [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_60.jdk/Contents/Home/bin/java (Nov 1, 2015, 12:32:04 PM)
*****
1997|DVD
L.A. Confidential
*****
1990|VHS
Pacific Heights
*****
2001|DVD
Amélie
A Beautiful Mind
Legally Blonde
The Lord of the Rings: The Fellowship of the Ring
Moulin Rouge
  
```

3.2 In the second test, assume the same table and index *yrInd ON Movie(year)*.

- 3.2.1 Find all movies made in 2000.

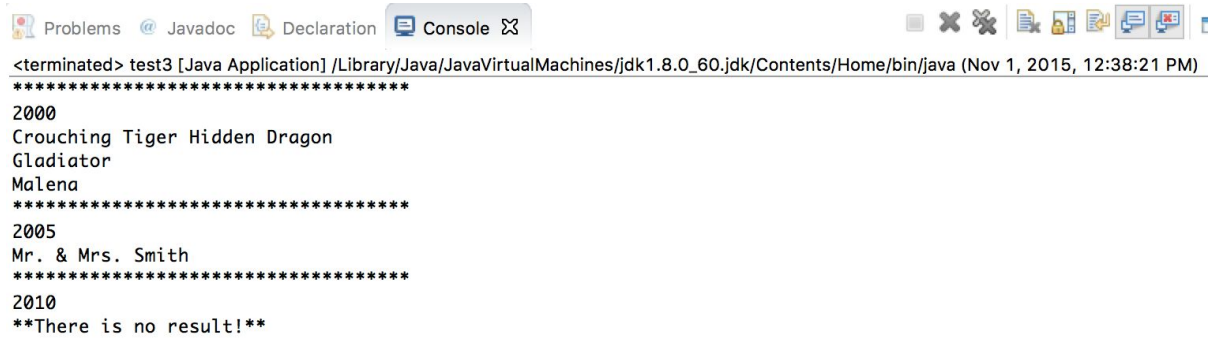
3.2.2 Find all movies made in 2005.

3.2.3 Find all movies made in 2010.

(1)method and process

- We insert the key and value by reading the file with split function, after insert all the elements, we search the movies made in 2000, 2005 and 2010.

(2)result



```
<terminated> test3 [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_60.jdk/Contents/Home/bin/java (Nov 1, 2015, 12:38:21 PM)
*****
2000
Crouching Tiger Hidden Dragon
Gladiator
Malena
*****
2005
Mr. & Mrs. Smith
*****
2010
**There is no result!**
```