```java
package cn.sxt.mycollection;
/**
 * 自定义实现一个ArrayList,体会底层原理
 *增加remove方法
 *共有size,isEmpty,add,get,set,remove方法
 * @author 江
 *
 */
public class SxtArrayList05<E> {
        private   Object[] elementData;
        private   int size;
        private static final int DEFAULT_CAPACITY=10;

        public SxtArrayList05() {
                elementData=new Object[DEFAULT_CAPACITY];
        }
        public SxtArrayList05(int capacity) {
                if(capacity<0) {
                        throw new RuntimeException("容器的容量不能为负数");
                }else if(capacity==0) {
                        elementData=new Object[DEFAULT_CAPACITY];
                }else {
                        elementData=new Object[capacity];
                }
        }

        public int size() {
                return size;
        }

        public boolean isEmpty() {
                return size==0? true:false;
        }

        public void add(E element) {
                //什么时候扩容?
                if(elementData.length==size) {
                        //扩容操作
                        Object[] newArray=new Object[elementData.length+
(elementData.length>>1)];
```

```java
                        System.arraycopy(elementData,0,newArray,0,elementData.length);
                        elementData=newArray;
                }
                elementData[size++]=element;
        }


        public E get(int index) {
                checkRange(index);
                return (E)elementData[index];
        }


        public void set(int index,E element) {

                checkRange(index);
                elementData[index]=element;
        }


        //索引合法判断【10，size）
        public void checkRange(int index) {
                if(index<0||index>size-1) {
                        //不合法
                        throw new RuntimeException("索引不合法："+index);
                }
        }


        public void remove(E element) {
                //element，将它和所有元素挨个比较，获得第一个比较为true的。返回
                for(int i=0;i<size;i++) {
                        if(element.equals(get(i))) {      //容器中所有的比较操作，都是
用equals而非==

                                remove(i);
                        }
                }
        }


        public void remove(int index) {
                int numMoved=elementData.length-index-1;
                if(numMoved>0) {
                System.arraycopy(elementData,index+1,elementData, index, numMoved);
                }
```

```java
            elementData[--size]=null;
    }



    @Override
    public String toString() {
            StringBuilder sb=new StringBuilder();
            sb.append('[');
            for(int i=0;i<size;i++) {
                sb.append(elementData[i]+",");
            }
            sb.setCharAt(sb.length()-1,']');

            return sb.toString();
    }

    public static void main(String[] args) {
                SxtArrayList05 s1=new SxtArrayList05(20);
                s1.add("aa");
                s1.add("bb");
                for(int i=0;i<40;i++) {
                        s1.add("jiang"+i);
                }

                s1.set(9, "jjjjj");
                System.out.println(s1);
                System.out.println(s1.get(10));
                System.out.println(s1.get(9));

                s1.remove(3);
                System.out.println(s1);

                System.out.println(s1.size());
                System.out.println(s1.isEmpty());
    }

}
```