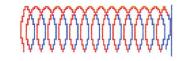# ArrayList

Doubles when reaches full capacity

"capacity" is the ability to store
"size" is the ability to actually store

When size == capacity, automatically gets double capacity

ArrayList <String> a = new ArrayList <String> (4);

```
0   1   2   3
NC  SC  GA
```

```
0   1   2   3
NC  SC  VA  ~~FL~~GA
a   TH
```

```
0   1   2   3
NC  TH  GA  FL  ←
```
← Technically GA
  capacity 3
  size 4

exactly unique deep copy
```
0   1   2   3   4   5   6   7     capacity 8
NC  TH  GA  FL  FL                  size  5
```

Output : 5
     NC, TH, GA, FL, FL

Using the **ArrayList** Class in Java

Method Name	Parameter Type	Returns	Operation Performed
ArrayList	(none)		Constructs an empty array list of capacity 10.
ArrayList	int		Constructs an empty array list of the capacity indicated by the parameter.
add	int, Object	void	Inserts the specified Object at the specified position; shifts all subsequent elements to the right one place.
add	Object	void	Inserts the specified Object at the end.
ensureCapacity	int	void	Increases the capacity of the array list to at least the specified capacity, if it is currently less than the specified capacity.
get	int	Object	Returns the element at the specified position.
isEmpty	(none)	boolean	Returns true if the array list is empty, false otherwise.
remove	int	Object	Removes the element at the specified position, shifts all subsequent elements to the left one place, and returns the removed element.
size	(none)	int	Returns current size.
trimToSize	(none)	void	Trims the capacity of the array list to its size.

```java
import java.util.*;
public class arrayList
{
        public static void main(String[] args)
        {
                ArrayList<String> a = new ArrayList<String>(4);
                a.add("NC");
                a.add("SC");
                a.add("GA");
                a.add(2, "VA");
                a.set(1, "TN");
                a.remove(a.indexOf("VA"));
                a.add("FL");
                a.add(a.get(a.size()-1));

                System.out.println(a.size());
                for (int i = 0;  i < a.size();  i++)
                        System.out.print(a.get(i) + " ");
                System.out.println();

                Object[] myArray = a.toArray();
                for (int i = 0;  i < myArray.length;  i++)
                        System.out.print(myArray[i]+ " ");
                System.out.println();
        }
}
```

Handwritten annotations:

Template Supports any class

All classes 'inherit' and default to String(), equals() and close() from this class

Superclass of all other classes