

# COMP 250

## INTRODUCTION TO COMPUTER SCIENCE

Assignment Project Exam Help

<https://powcoder.com>

Week 5-1: Arraylist

Giulia Alberini, Fall 2020

# WHAT ARE WE GOING TO DO IN THIS VIDEO?



- **Arraylist**

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Assignment Project Exam Help

**ARRAY LISTS**  
<https://powcoder.com>

Add WeChat powcoder

# ARRAYS IN JAVA

- Arrays whose elements have a primitive type

```
int[] myInts = new int[15];  
myInts[3] = -732;
```

Assignment Project Exam Help

<https://powcoder.com>

- Arrays whose elements have a reference type

```
Shape[] myShapes = new Shape[428];  
shapes[293] = new Shape(▲);
```

Add WeChat powcoder

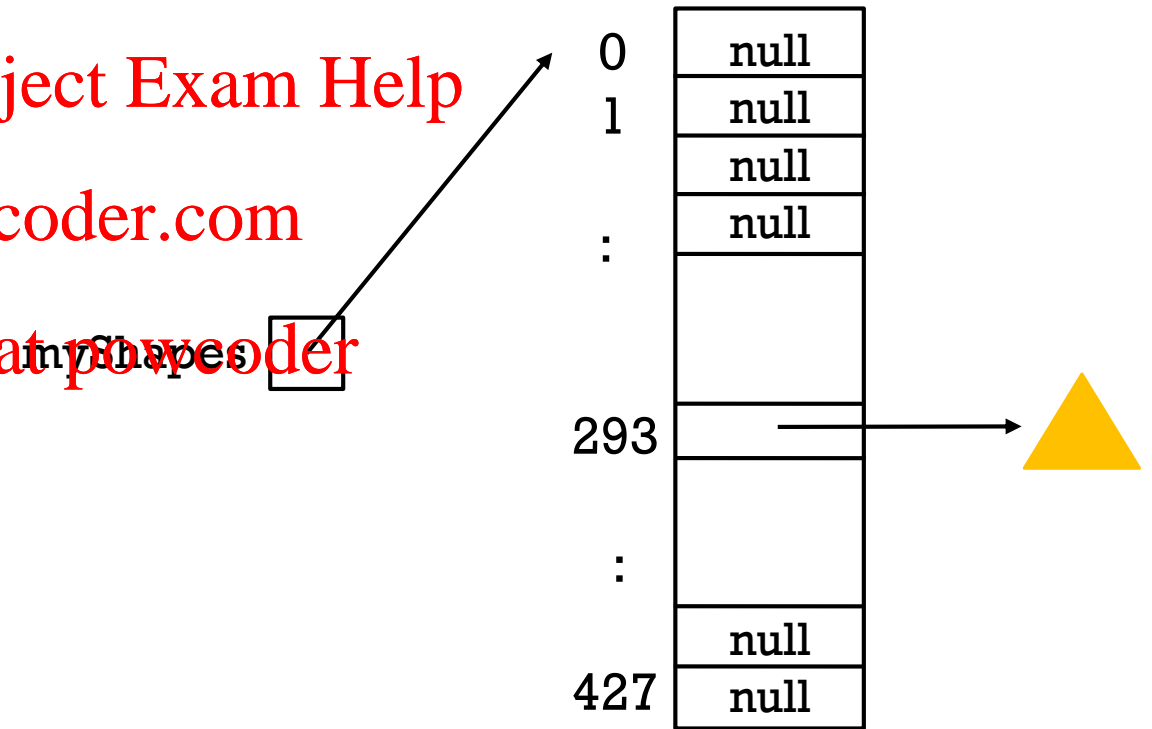
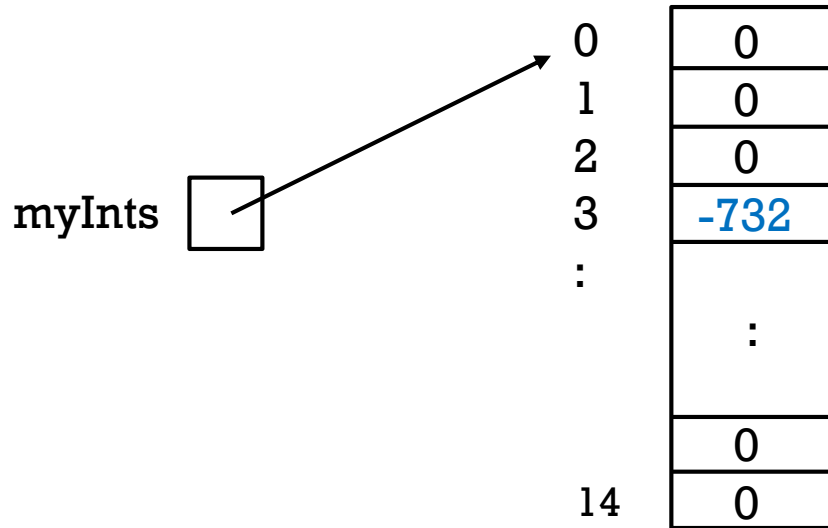
```
int[] myInts = new int[15];  
myInts[3] = -732;
```

```
Shape[] myShapes = new Shape[428];  
shapes[293] = new Shape(▲);
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



You can think of an array as a block of consecutive slots in memory

## ARRAYS HAVE CONSTANT TIME ACCESS

A computer accesses an element in an array in constant time

**Assignment Project Exam Help**

i.e. constant, independent of the length  $N$  of the array.

<https://powcoder.com>

```
.... = a[k] ; // read  
a[k] = .... ; // write
```

**Add WeChat powcoder**

You will learn more about how this works in COMP 206 and 273.

# LIST

---

An ordered set of elements

Assignment Project Exam Help

$a_0, a_1, a_2, \dots, a_{N-1}$   
<https://powcoder.com>

Add WeChat powcoder

$N$  is the number of elements in the list, often called the “size” of the list.

## WHAT WOULD WE LIKE TO DO WITH A LIST?

<code>get(i)</code>	<code>// Returns the i-th element (but doesn't remove it)</code>
<code>set(i,e)</code>	<code>// Replaces the i-th element with e</code>
<code>add(e)</code>	<code>// Append element e at the end of the list</code>
<code>add(i,e)</code>	<code>// Inserts element e into the i-th position</code>
<code>remove(i)</code>	<code>// Removes the i-th element from list</code>
<code>remove(e)</code>	<code>// Removes first occurrence of element e // from the list (if it is there)</code>
<code>clear()</code>	<code>// Empties the list.</code>
<code>isEmpty()</code>	<code>// Returns true if empty, false if not empty.</code>
<code>size()</code>	<code>// Returns number of elements in the list</code>

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



# IMPLEMENTATIONS

There are different implementations of a list:

- Array list      Assignment Project Exam Help
- Singly linked list      <https://powcoder.com>  
next two videos
- Doubly linked list      Add WeChat powcoder

# IMPLEMENTATIONS

There are different implementations of a list:

- Array list → [Assignment Project Exam Help](#)
- Singly linked list <https://powcoder.com>
- Doubly linked list [Add WeChat powcoder](#)

# ARRAYLIST

## Idea:

- Use an array to store the elements of the list
- Keep track of how many elements we have inserted in the list

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder:

*ArrayList.java*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
}
```

## To decide:

- How big should the underlying array be when we first create an object of type ArrayList?  
(this is referred to as the *initial capacity* of the list)

# ARRAYLIST

## Idea:

- Use an array to store the elements of the list
- Keep track of how many elements we have inserted in the list

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

## To decide:

- How big should the underlying array be when we first create an object of type ArrayList?  
→ Java's ArrayList creates an array of length 10.

*ArrayList.java*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    public ArrayList() {  
        arr = new Shape[10];  
        size = 0;  
    }  
}
```

## EXAMPLE

*ArrayList.java*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    public ArrayList() {  
        arr = new Shape[10];  
        size = 0;  
    }  
}
```

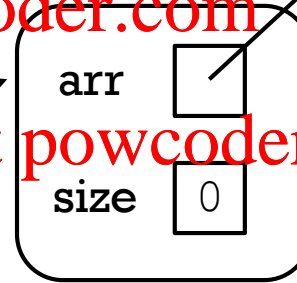
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
ArrayList list = new ArrayList();
```

list



0	null
1	null
2	null
3	null
4	null
5	null
6	null
7	null
8	null
9	null

## EXAMPLE – WHAT WE WANT WHEN ADDING ELEMENTS

*ArrayList.java*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    public ArrayList() {  
        arr = new Shape[10];  
        size = 0;  
    }  
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
ArrayList list = new ArrayList();  
// add 7 elements...
```

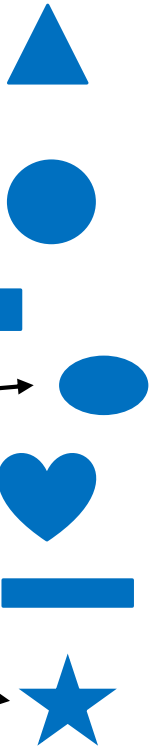
list

arr  
size

7

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

null  
null  
null  
null  
null  
null  
null  
null  
null  
null



# HOW TO IMPLEMENT VARIOUS OPERATIONS? – get ()

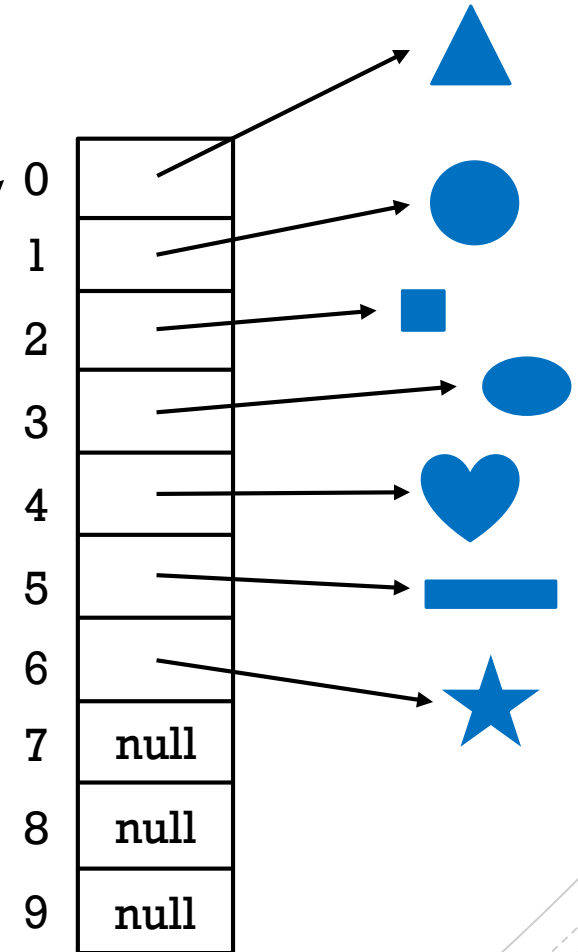
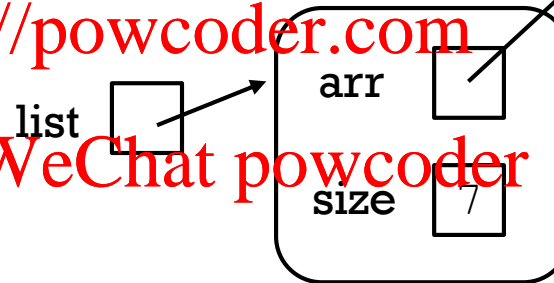
*Returns the element at the specified position in this list.*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    :  
  
    public Shape get(int i) {  
        if(  
            )  
            return arr[i];  
        // otherwise?  
    }  
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



# HOW TO IMPLEMENT VARIOUS OPERATIONS? – get ()

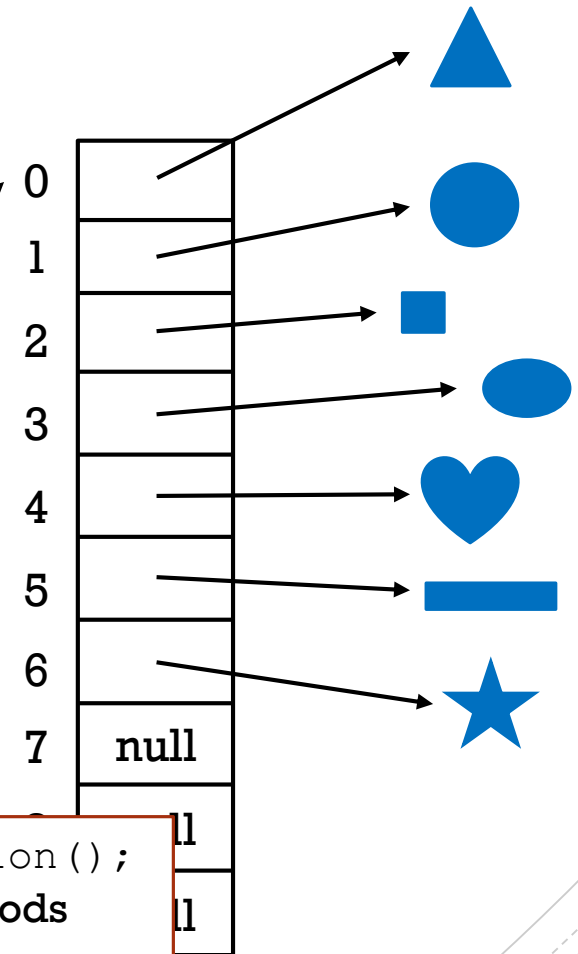
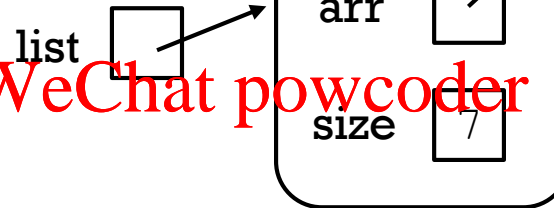
*Returns the element at the specified position in this list.*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    :  
  
    public Shape get(int i) {  
        if( i >= 0 && i < size )  
            return arr[i];  
        // otherwise?  
    }  
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



throw new IndexOutOfBoundsException();  
I will not mention this for all the other methods today, but it should be added for proper implementations.



# HOW TO IMPLEMENT VARIOUS OPERATIONS? – set ()

*Replaces the element at the specified position in this list with the specified element.*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    :  
  
    public Shape set(int i, Shape e)  
    {  
        if(i >= 0 && i < size) {  
            Shape tmp = arr[i];  
            arr[i] = e;  
            return tmp;  
        }  
    }  
}
```

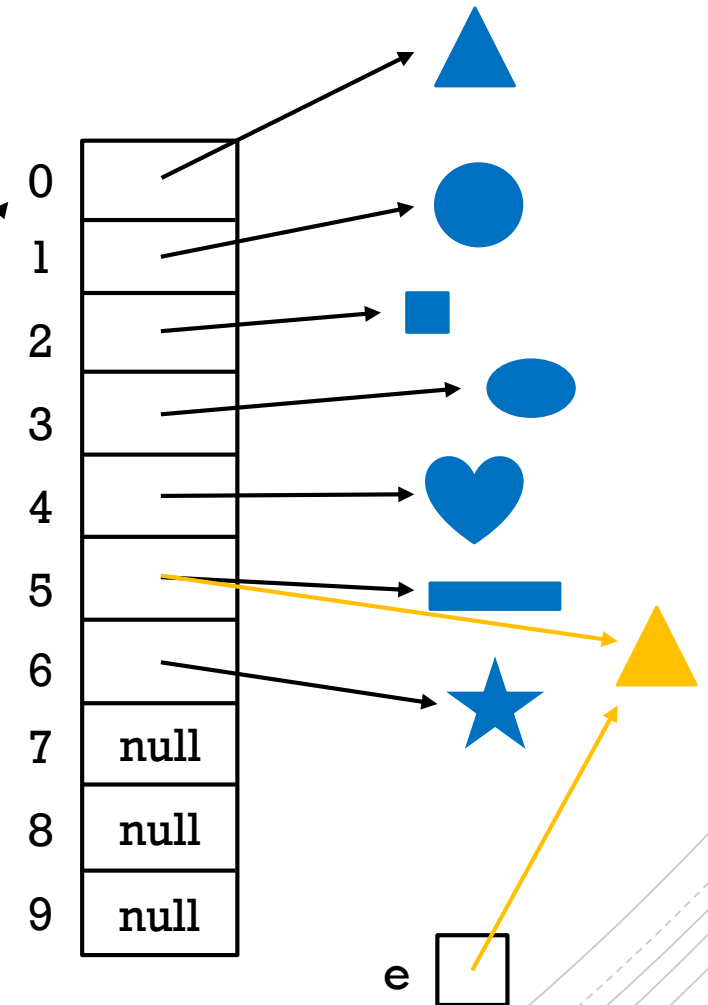
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

For example:

```
list.set(5, e);
```



# HOW TO IMPLEMENT VARIOUS OPERATIONS? – add ()

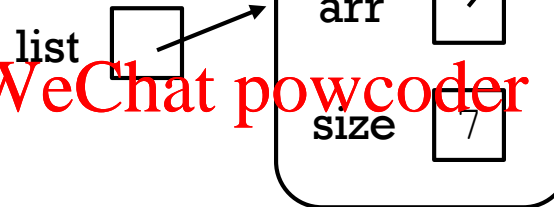
*Appends the specified element to the end of this list.*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    :  
  
    public void add(Shape e)  
    {  
        arr[    ] = e;  
        size = size + 1;  
    }  
}
```

Assignment Project Exam Help

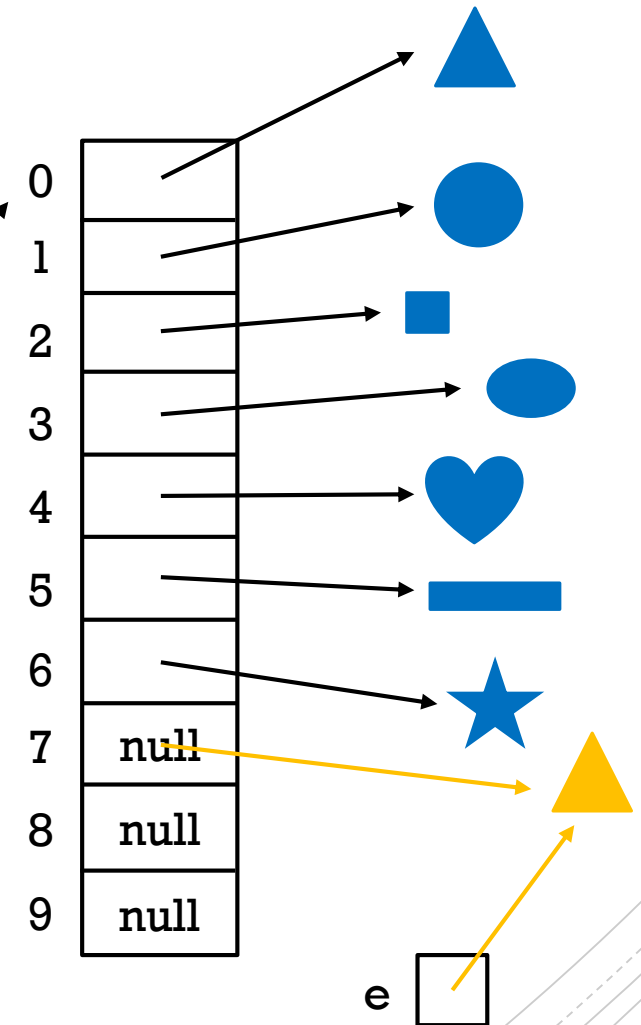
<https://powcoder.com>

Add WeChat powcoder



For example:  
`list.add(e);`

What if the array `arr` is full?



# HOW TO IMPLEMENT VARIOUS OPERATIONS? – add ()

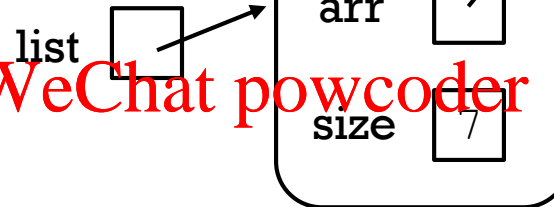
*Appends the specified element to the end of this list.*

```
public class ArrayList {  
    private Shape[] arr;  
    private int size;  
  
    :  
  
    public void add(Shape e)  
    {  
        arr[size] = e;  
        size = size + 1;  
    }  
}
```

Assignment Project Exam Help

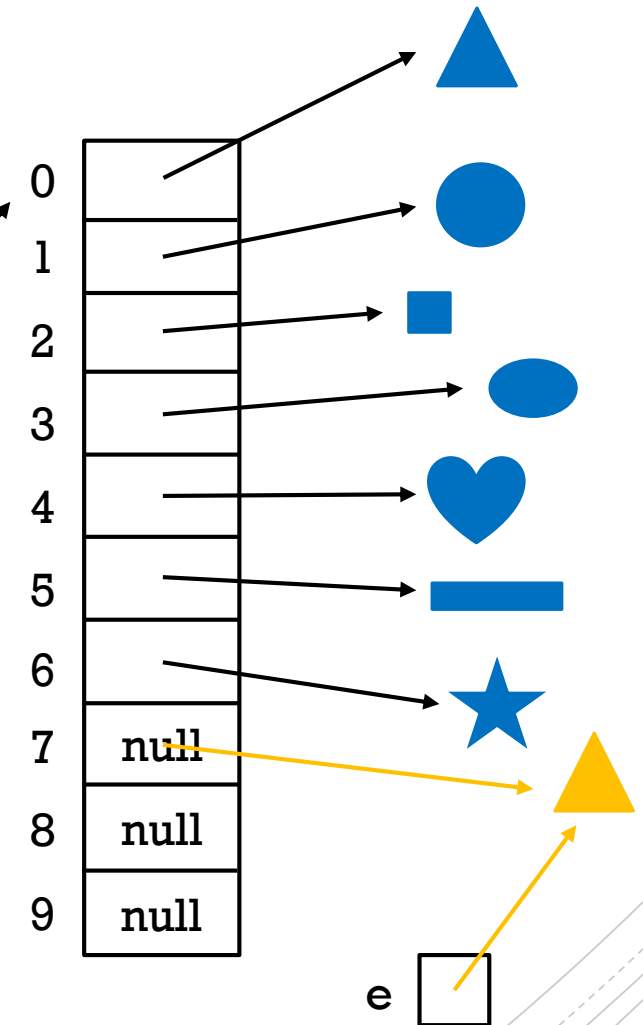
<https://powcoder.com>

Add WeChat powcoder



For example:  
`list.add(e);`

What if the array `arr` is full?



## HOW TO IMPLEMENT VARIOUS OPERATIONS? – add ()

*Appends the specified element to the end of this list.*

What if the array arr is already full?

```
public void add(Shape e) {  
    if ( Assignment Project Exam Help  
        resize();  
    arr[size] = e;  
    size = size + 1;  
}  
  
https://powcoder.com  
Add WeChat powcoder  
  
private void resize() {  
    Shape[] bigger = new Shape[arr.length*2]; // example  
    for(int i=0; i < size; i++) {  
        bigger[i] = arr[i];  
    }  
    arr = bigger;  
}
```

## HOW TO IMPLEMENT VARIOUS OPERATIONS? – add ()

*Appends the specified element to the end of this list.*

What if the array arr is already full?

```
public void add(Shape e) {  
    if (arr.length == size)  
        resize();  
    arr[size] = e;  
    size = size + 1;  
}
```

```
private void resize() {  
    Shape[] bigger = new Shape[arr.length*2]; // example  
    for(int i=0; i < size; i++) {  
        bigger[i] = arr[i];  
    }  
    arr = bigger;  
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

## OVERLOADING

`add( e )`      `// inserts element e at end of list`

`add( i ,e)`      `// Inserts element e into the i-th position`

<https://powcoder.com>

Add WeChat powcoder

`remove(i)`      `// Removes the i-th element from list`

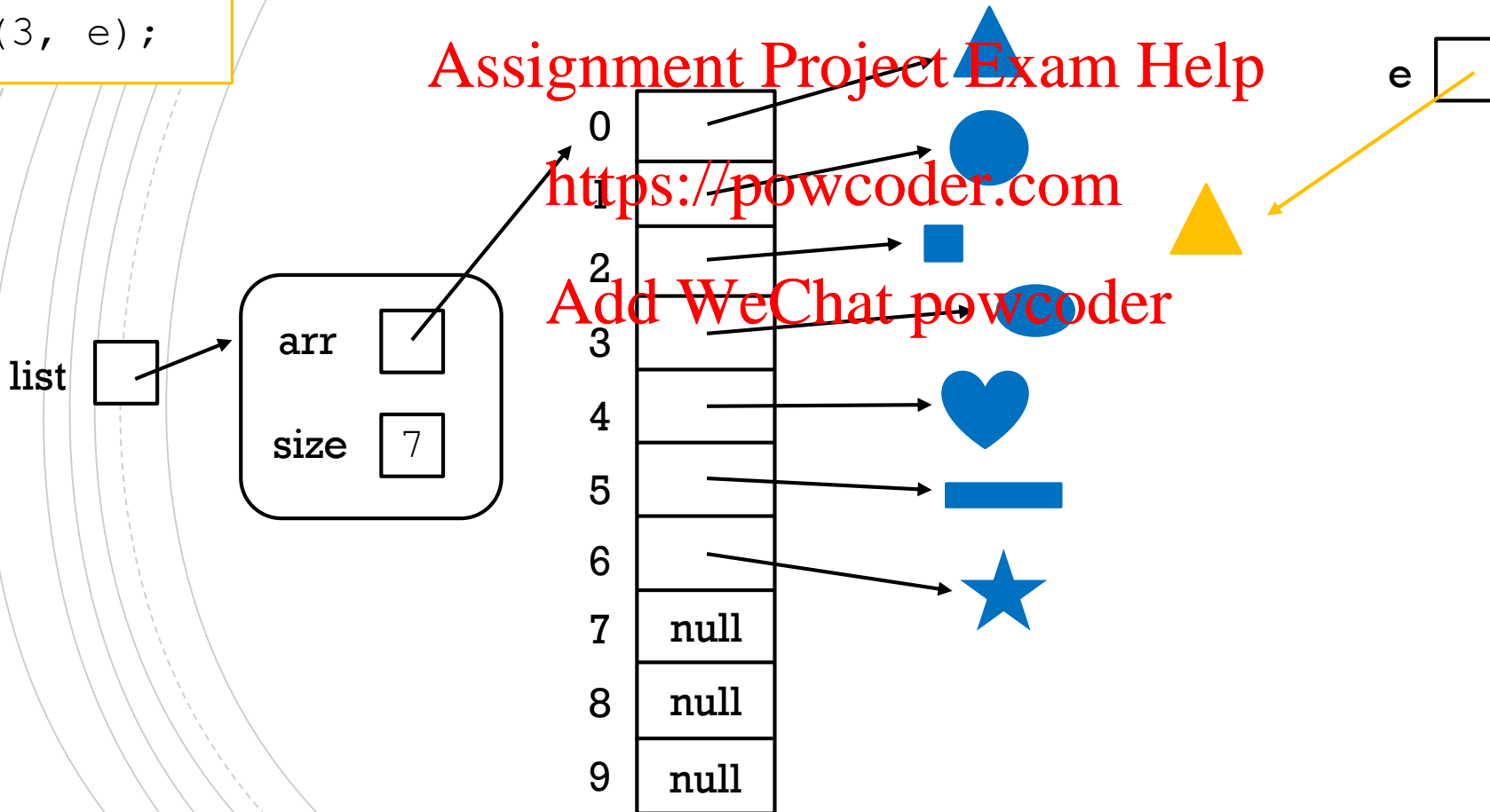
`remove(e)`      `// Removes first occurrence of element e`  
`// from the list (if it is there)`

# HOW TO IMPLEMENT `add(i, e)`

*Inserts the specified element at the specified position in the list.*

For example:

```
list.add(3, e);
```



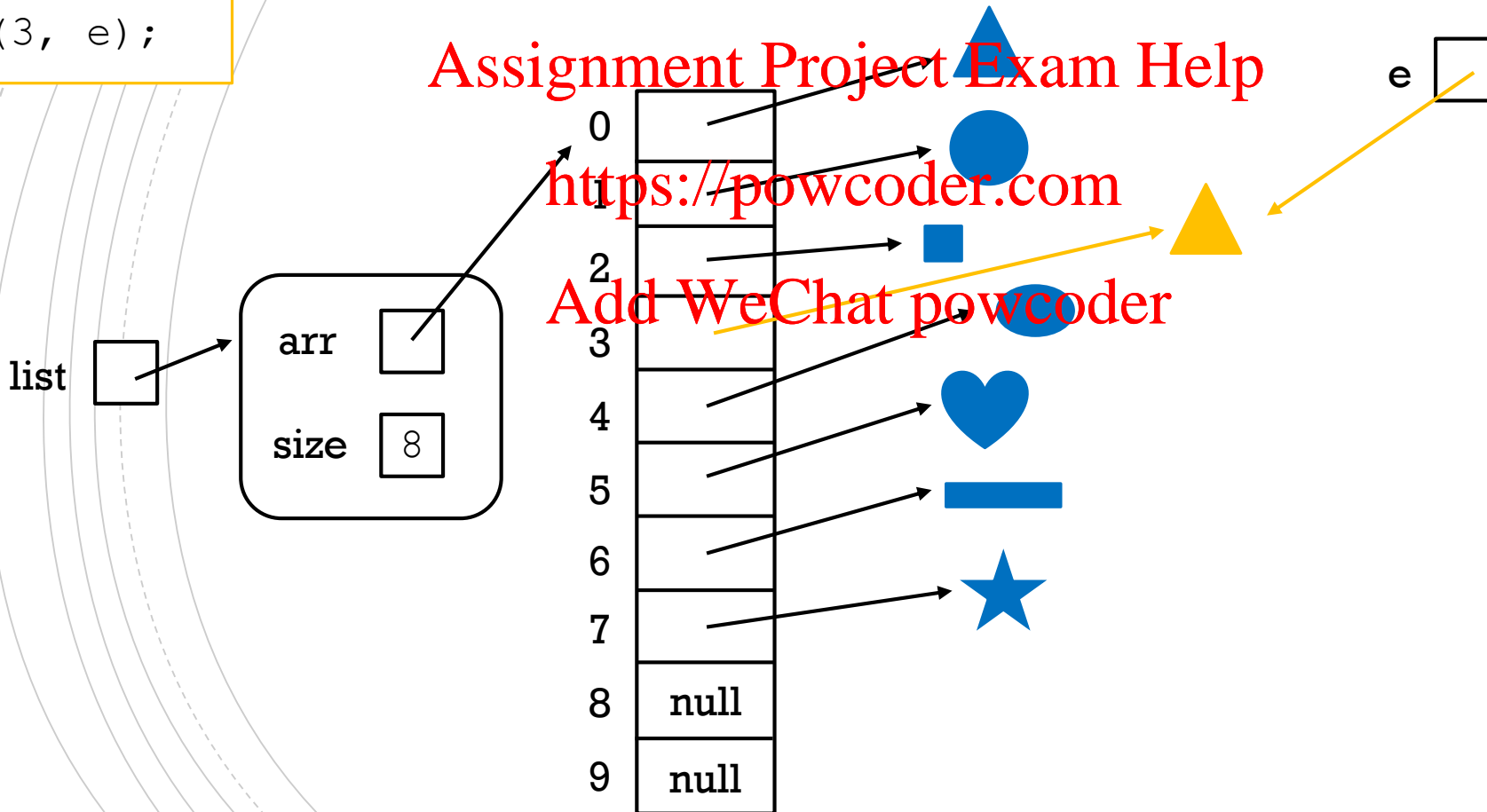
# HOW TO IMPLEMENT `add(i, e)`

*Inserts the specified element at the specified position in the list.*

IDEA: make room by shifting the elements, then add `e`

For example:

```
list.add(3, e);
```





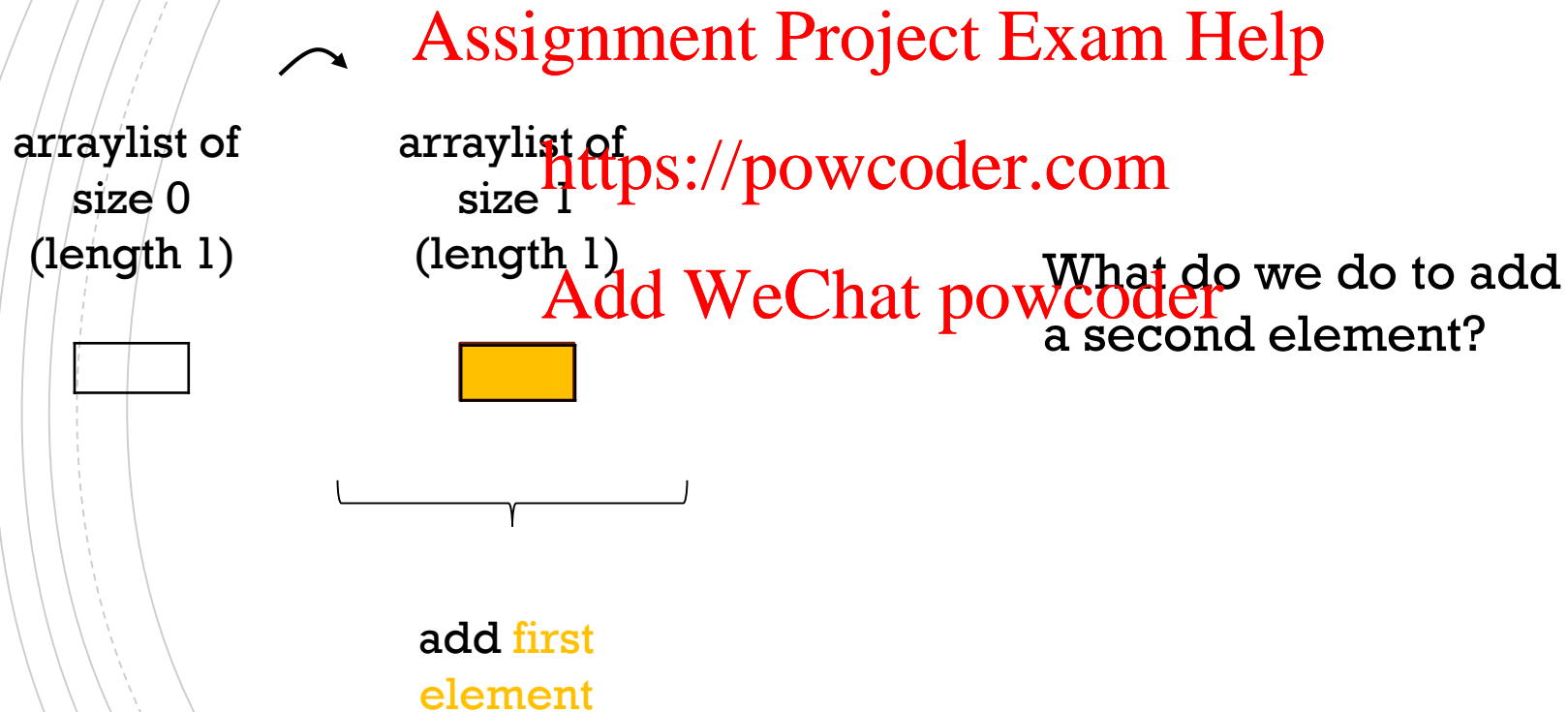
## HOW TO IMPLEMENT add(i,e)

*Inserts the specified element at the specified position in the list.*

```
public void add(int i, Shape e) {  
    // Throw exception if i is out of bounds  
    // Resize if not enough space  
    // Shift elements down  
    // Add the new element  
}
```

# ADDING N ELEMENTS TO AN ARRAY LIST

Suppose we initialize an array list with an empty array of length 1.  
We then add an element.



# ADDING N ELEMENTS TO AN ARRAY LIST

Suppose each time we add to a full array list, we double the length of the array.

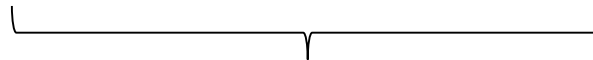
arraylist of  
size 1  
(length 1)



arraylist of  
size 1  
(length 2)



arraylist of  
size 2  
(length 2)



add **second**  
**element**

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# ADDING N ELEMENTS TO AN ARRAY LIST

Assignment Project Exam Help

arraylist of  
size 2  
(length 2)



arraylist of  
size 2  
(length 4)



arraylist of  
size 3  
(length 4)



<https://powcoder.com>

Add WeChat powcoder



add third element

# ADDING N ELEMENTS TO AN ARRAY LIST

Assignment Project Exam Help  
arraylist of size 3 (length 4) → arraylist of size 4 (length 4)  
<https://powcoder.com>

Add WeChat powcoder



add fourth  
element

## ADDING N ELEMENTS TO AN ARRAY LIST

arraylist of  
size 4  
(length 4)



arraylist of  
size 4  
(length 8)



arraylist of  
size 5  
(length 8)



Assignment Project Exam Help

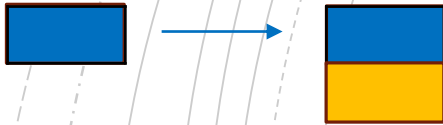
<https://powcoder.com>

Add WeChat powcoder

add fifth element

# ADDING N ELEMENTS TO AN ARRAY LIST

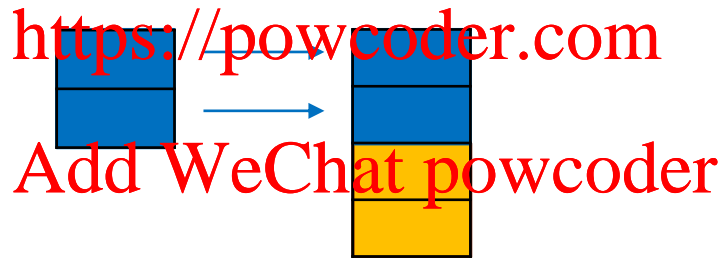
Double length and  
copy one element



add two elements

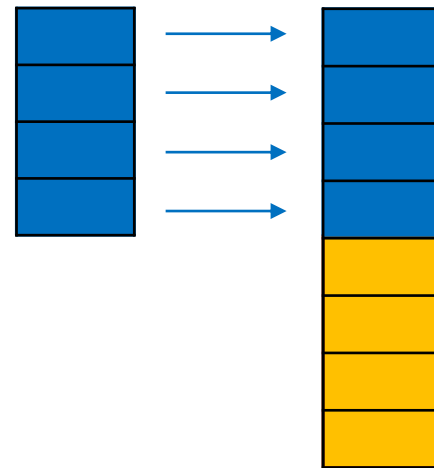
Double length and  
copy two elements

Assignment Project Exam Help



add 3-4 elements

Double length and  
copy four elements



add 5-8 elements

<https://powcoder.com>  
Add WeChat powcoder

Q: How many times  $k$  do we need to double the length of the array so that it is of length  $N$  ?

A:

Assignment Project Exam Help

<https://powcoder.com>

Q: How many **copy operations** are required to add  $N$  elements to an empty array list ?

Add WeChat powcoder

A:



Q: How many times  $k$  do we need to double the length of the array so that it is of length  $N$  ?

A:  $2 \cdot 2 \cdots 2 = 2^k = N \rightarrow k = \log_2 N$   
Assignment Project Exam Help

<https://powcoder.com>  
Q: How many **copy operations** are required to add  $N$  elements to an empty array list ?  
Add WeChat powcoder

A:

Q: How many times  $k$  do we need to double the length of the array so that it is of length  $N$  ?

A:  $2 \cdot 2 \cdots 2 = 2^k = N \rightarrow k = \log_2 N$   
Assignment Project Exam Help

<https://powcoder.com>

Q: How many **copy operations** are required to add  $N$  elements to an empty array list ?  
Add WeChat powcoder

A:  $1 + 2 + 4 + \dots + 2^{k-1}$

## SERIES

- *Geometric series*

$$\sum_{i=0}^n x^i = 1 + x + x^2 + \cdots + x^n = \frac{1 - x^{n+1}}{1 - x}$$

<https://powcoder.com>

Add WeChat powcoder

- *Arithmetic series*

$$\sum_{i=1}^n i = 1 + 2 + \cdots + n = \frac{1}{2}n(n + 1)$$

Q: How many times  $k$  do we need to double the length of the array so that it is of length  $N$  ?

A:  $2 \cdot 2 \cdots 2 = 2^k = N \rightarrow k = \log_2 N$   
Assignment Project Exam Help

<https://powcoder.com>

Q: How many **copy operations** are required to add  $N$  elements to an empty array list ?  
Add WeChat powcoder

A:  $1 + 2 + 4 + \dots + 2^{k-1} = \sum_{i=0}^{k-1} 2^i = \frac{1-2^k}{1-2} = 2^k - 1 = N - 1$

# JAVA ARRAYLIST CLASS

<https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>

## Assignment Project Exam Help

- It uses an array as the underlying data structure

<https://powcoder.com>

- It grows the array (by 50%, ~~not 100%~~) when the array is full and a new element is added.
- As a client, we don't have access to the fields directly. We need to use the methods provided (get(), set(), ...) to manipulate the list.

# JAVA ARRAYLIST – GENERIC CLASS

- ArrayList is a generic class with a type parameter.
- When you create an object of type ArrayList you specify the type of the elements stored by the list by appending to ArrayList a class name enclosed in angle brackets.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- Example:

```
// creates an arraylist of integers with initial capacity 10
ArrayList<Integer> words = new ArrayList<Integer>();

// creates an arraylist of shapes with initial capacity 23
ArrayList<Shape> myShapes = new ArrayList<Shape>(23);
```

If we write `int` instead we get a compile-time error!

## WRAPPER CLASSES

- Integer, Double, and Character wrap a value of the primitive type `int`, `double`, and `char` (respectively) in an object. Thus, they turn primitive types into reference types.

Assignment Project Exam Help

<https://powcoder.com>

- The conversion between the primitive types and their wrappers is done automatically. For example, the following would not cause a compile-time error:

Add WeChat powcoder

```
Integer x = 5;
```

- Note that these classes have static methods/attributes that you might have already used. For example: `Integer.MAX_VALUE`, `Double.parseDouble()`

# AUTOBOXING AND UNBOXING

- *Autoboxing* is the automatic conversion that the Java compiler makes between the primitive types and their corresponding object wrapper classes. For example converting an `int` to an `Integer`.

<https://powcoder.com>

```
Integer x = 5;
```

Add WeChat powcoder

- If the conversion goes the other way, this is called *unboxing*.

```
Integer x = new Integer(5);  
int y = x;
```

- <https://docs.oracle.com/javase/tutorial/java/data/autoboxing.html>



## IMMUTABLE TYPES

- Note that `Integer`, `Double`, and `Character` are **immutable** reference types (like `String`).  
<https://powcoder.com>
- As with `String`, you can appear to update values, but you are never changing the actual Object. A new Object gets created each time we “change” a value.

## WHY WRAPPER CLASSES?

Assignment Project Exam Help

It is much simpler (in terms of code re-use) to have ArrayList require the input to be an Object (a reference type), instead of using primitive types.

<https://powcoder.com>  
Add WeChat powcoder

For example, all reference types can be compared using `.equals()`, while we have to use `==` for primitive types.

# THE FOREACH LOOP

```
int[] numbers = {1, 2, 3, 4, 5};  
for(int element: numbers) {  
    System.out.println(element);  
}
```

The foreach loop (also called enhanced for loop) can make your code more readable and can be convenient to use. It is not helpful when you need to refer to the index of an element.



# Coming Soon

## Assignment Project Exam Help

In the next videos:

- Linked lists <https://powcoder.com>

Add WeChat powcoder