Lab Manual

CL2001 – Data Structures

Fall-2023

National University of Computer and Emerging Sciences-FAST
Karachi Campus

Data Structures Lab#1

Course: Data Structures (CL2001) **Semester:** Fall 2023
Instructor: Shafique Rehman **T.A:** N/A

Note:

- Maintain discipline during the lab.
- Listen and follow the instructions as they are given.
- Just raise hand if you have any problem.
- Completing all tasks of each lab is compulsory.
- Get your lab checked at the end of the session.

Content:

- Introduction of Command Line Argument
- Multidimensional Array
- Jagged Arrays
- Dynamic Arrays

Java Command Line Arguments

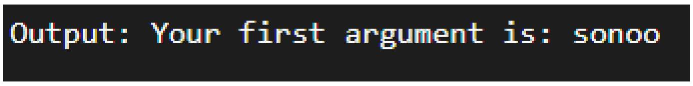
1. Command Line Argument
2. Simple example of command-line argument

The java command-line argument is an argument i.e. passed at the time of running the java program.

The arguments passed from the console can be received in the java program and it can be used as an input.

Simple example of command-line argument in java

```
public class Main {
  public static void main(String[] args) {
    System.out.println("Your first argument is: "+args[0]);
  }
}
```



```
Output: Your first argument is: sonoo
```

Hint: ParseInt is builtin method used to convert String values in IntegerSyntax: int i=Integer.parseInt(stringValue

Multidimensional Array in Java

Syntax to Declare Multidimensional Array in Java

```
dataType[][]arrayRefVar;  (or)
dataTypearrayRefVar[][];  (or)
dataType[]arrayRefVar[];
```

Jagged Array in Java

If we are creating odd number of columns in a 2D array, it is known as a jagged array. In other words, it is an array of arrays with different numberof columns.

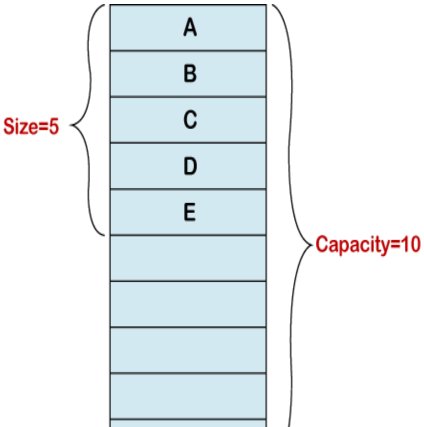
Syntax:

```
Datatype arr_name[][] = new
datatype[size][];
```

Dynamic Array in Java

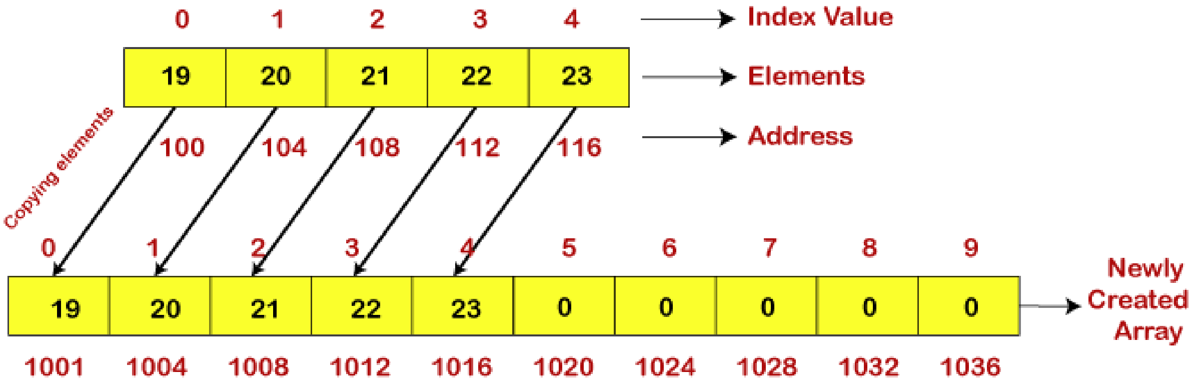
An array is a fixed size, homogeneous data structure. The limitation of arrays is that they're fixed in size. It means that we must specify the number of elements while declaring the array. Here a question arises that what if we want to insert an element and there is no more space is left for the new element? Here, the concept of dynamicarray comes into existence. It expends the size of the array dynamically.

Size and Capacity

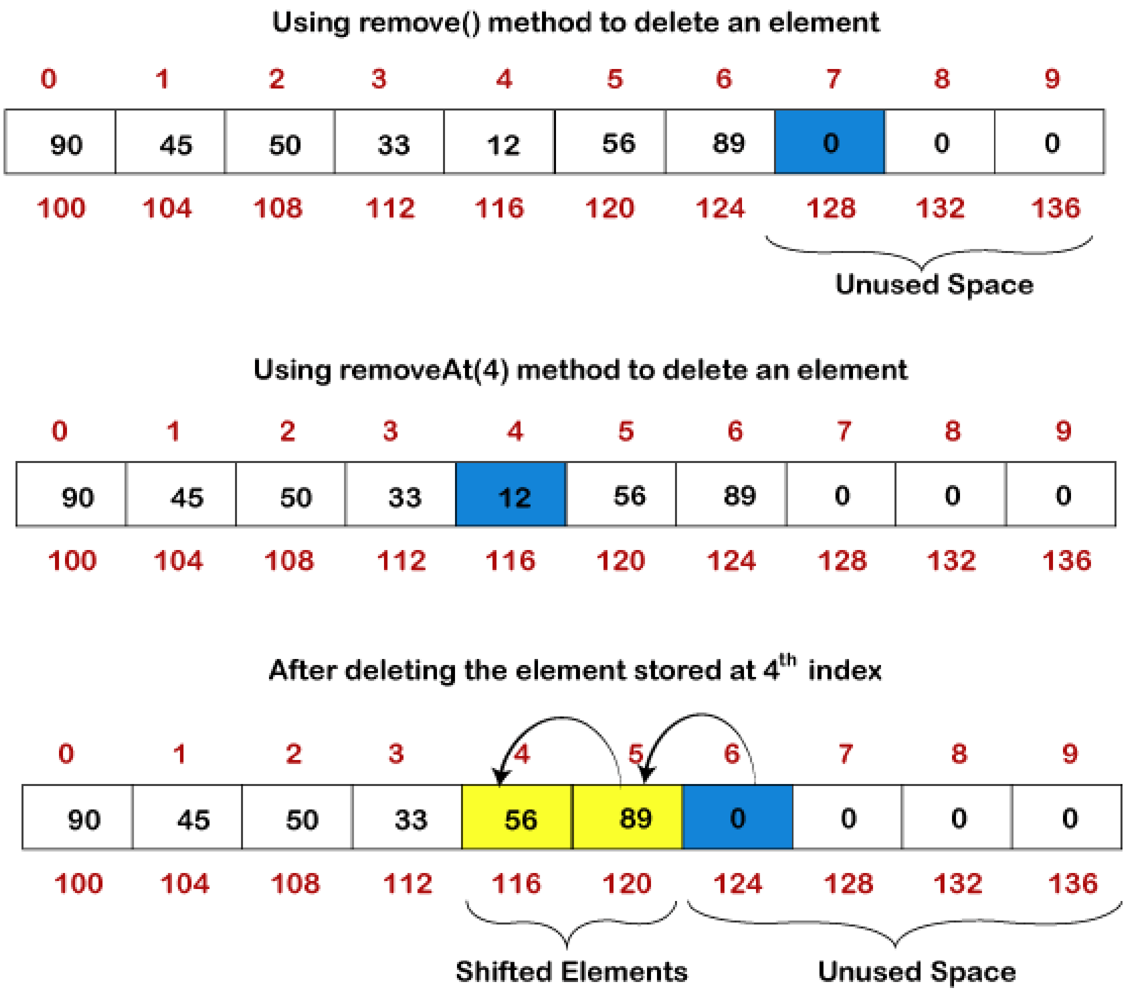


Features of Dynamic Array

Add Element in a Dynamic Array



Delete an Element from a Dynamic Array



Using remove() method to delete an element

0	1	2	3	4	5	6	7	8	9
90	45	50	33	12	56	89	0	0	0
100	104	108	112	116	120	124	128	132	136

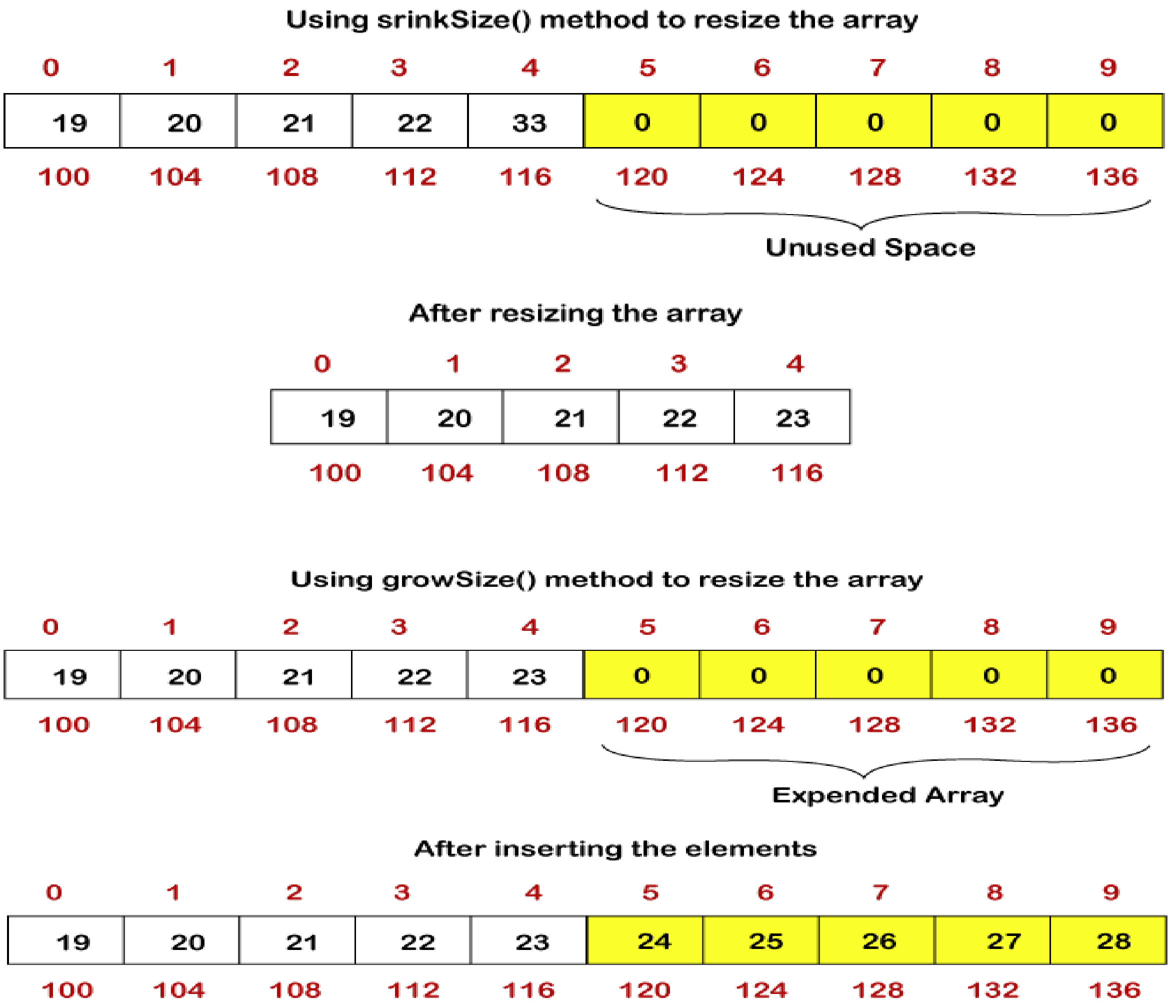
Using removeAt(4) method to delete an element

0	1	2	3	4	5	6	7	8	9
90	45	50	33	12	56	89	0	0	0
100	104	108	112	116	120	124	128	132	136

After deleting the element stored at 4th index

0	1	2	3	4	5	6	7	8	9
90	45	50	33	56	89	0	0	0	0
100	104	108	112	116	120	124	128	132	136

Resizing a Dynamic Array in Java



Using srinkSize() method to resize the array

0	1	2	3	4	5	6	7	8	9
19	20	21	22	33	0	0	0	0	0
100	104	108	112	116	120	124	128	132	136

Unused Space

After resizing the array

0	1	2	3	4
19	20	21	22	23
100	104	108	112	116

Using growSize() method to resize the array

0	1	2	3	4	5	6	7	8	9
19	20	21	22	23	0	0	0	0	0
100	104	108	112	116	120	124	128	132	136

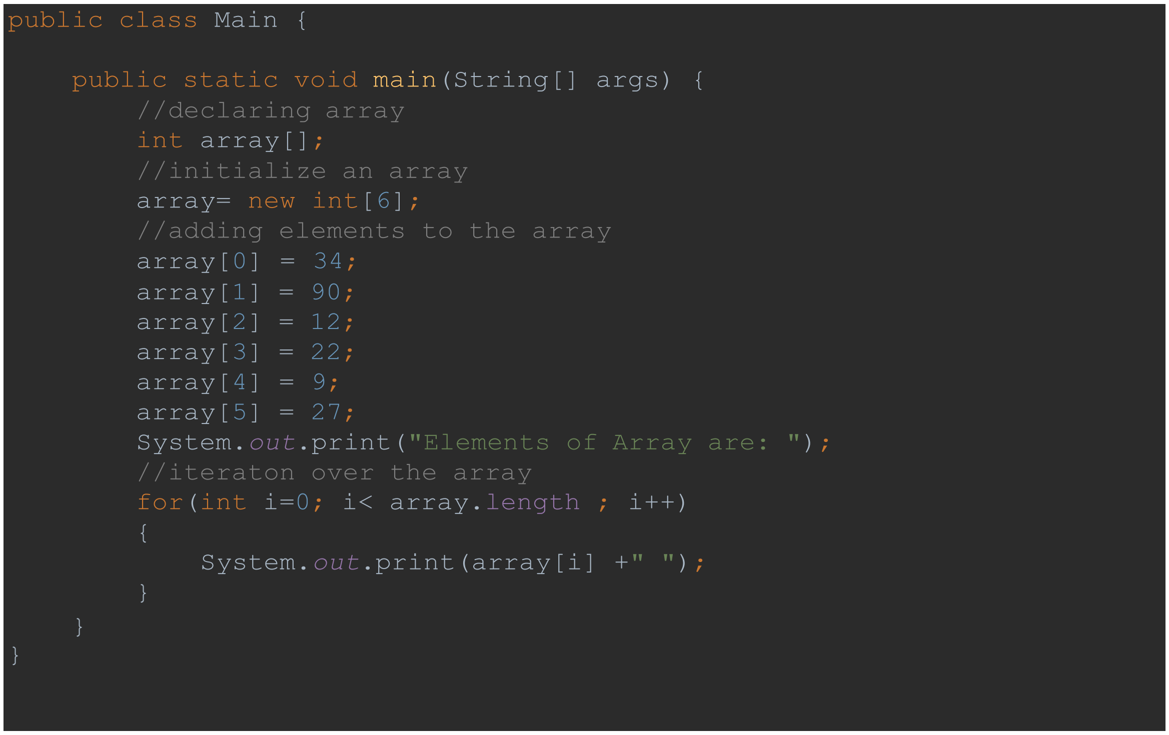
Expended Array

After inserting the elements

0	1	2	3	4	5	6	7	8	9
19	20	21	22	23	24	25	26	27	28
100	104	108	112	116	120	124	128	132	136

Initialize a Dynamic Array

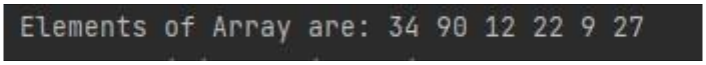
The initialization of the dynamic array is the same as the static array. Consider the following Java program that initializes a dynamicarray.



```java
public class Main {

    public static void main(String[] args) {
        //declaring array
        int array[];
        //initialize an array
        array= new int[6];
        //adding elements to the array
        array[0] = 34;
        array[1] = 90;
        array[2] = 12;
        array[3] = 22;
        array[4] = 9;
        array[5] = 27;
        System.out.print("Elements of Array are: ");
        //iteraton over the array
        for(int i=0; i< array.length ; i++)
        {
            System.out.print(array[i] +" ");
        }
    }
}
```

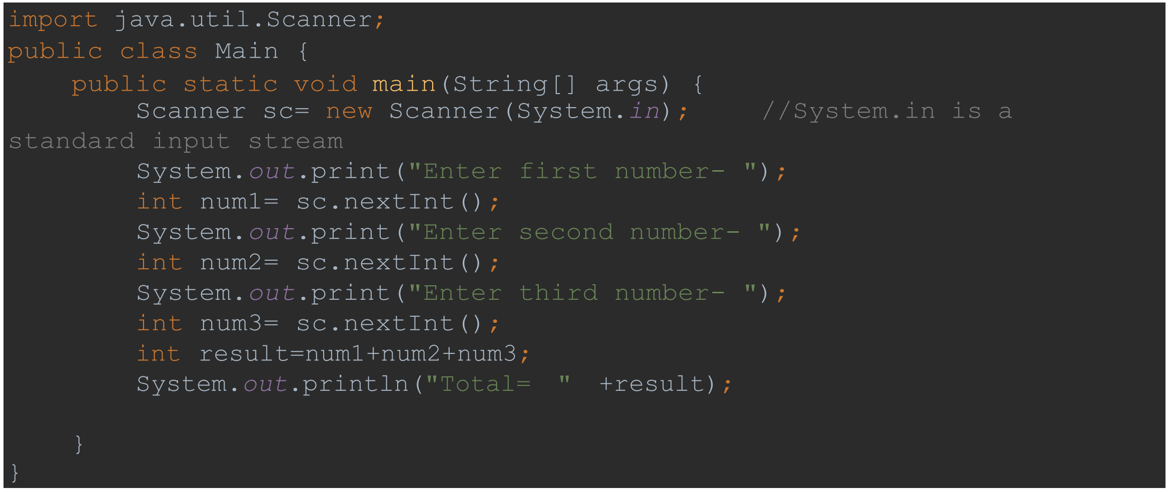
Output:



```
Elements of Array are: 34 90 12 22 9 27
```

It also converts the Bytes (from the input stream) into characters using the platform's default charset.

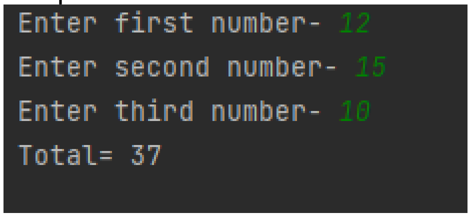
Example:



```
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner sc= new Scanner(System.in);     //System.in is a
standard input stream
        System.out.print("Enter first number- ");
        int num1= sc.nextInt();
        System.out.print("Enter second number- ");
        int num2= sc.nextInt();
        System.out.print("Enter third number- ");
        int num3= sc.nextInt();
        int result=num1+num2+num3;
        System.out.println("Total=  "  +result);

    }
}
```

Output:



```
Enter first number- 12
Enter second number- 15
Enter third number- 10
Total= 37
```

Implementation of

1. growSize()
2. shrinkSize()

1. //implementation of growSize()

```
public void growSize()
{
//declares a temp[] array
int temp[] = null;
if (count == sizeofarray)
{
//initialize a double size array of array
temp = new int[sizeofarray * 2];
{
for (int i = 0; i < sizeofarray; i++)
{
```

```
//copies all the elements of the old array
temp[i] = array[i];
}
}
}
array = temp;
sizeofarray= sizeofarray * 2;
}
```

2. **//Implementation of shrinkSize()**

   ```
   public void shrinkSize()
   {
   //declares a temp[] array
   int temp[] = null;
   if (count > 0)
   {
   //creates an array of the size equal to the count i.e. number of elements te array have
   temp = new int[count];
   for (int i = 0; i < count; i++)
   {
   //copies all the elements of the old array
   temp[i] = array[i];
   }
   sizeofarray = count;
   array = temp;
   }
   }
   ```

Exercise Lab 1:

Task#1: Using Command Line prints all the values which are given at run time.

Task#2: Write a Java function to rotate an array to the right by a given number of steps. For example, if the array is [1, 2, 3, 4, 5] and the number of steps is 2, the rotated array should be [4, 5, 1, 2, 3].

Task#3: Declare and initialize two the 2D array and print the Addition of those arrays.

Task#4: Declare and initialize Jagged array and return the sum of all indexes.

Task#5: Consider the dynamic array of size 4: elements are 10, 11, 14, 16. Now add the element at the end and print the updated array. Now remove the two elements and then print the updated array. Do not use another array.

Task#6: Declare the Dynamic array and ask the user to give 6 names of fruits and any name ofperson randomly. Now, replace the name of person with "orange". And print the updated array.