Fundamentals of Object Oriented Programming

CSN- 103

Dr. R. Balasubramanian

Associate Professor

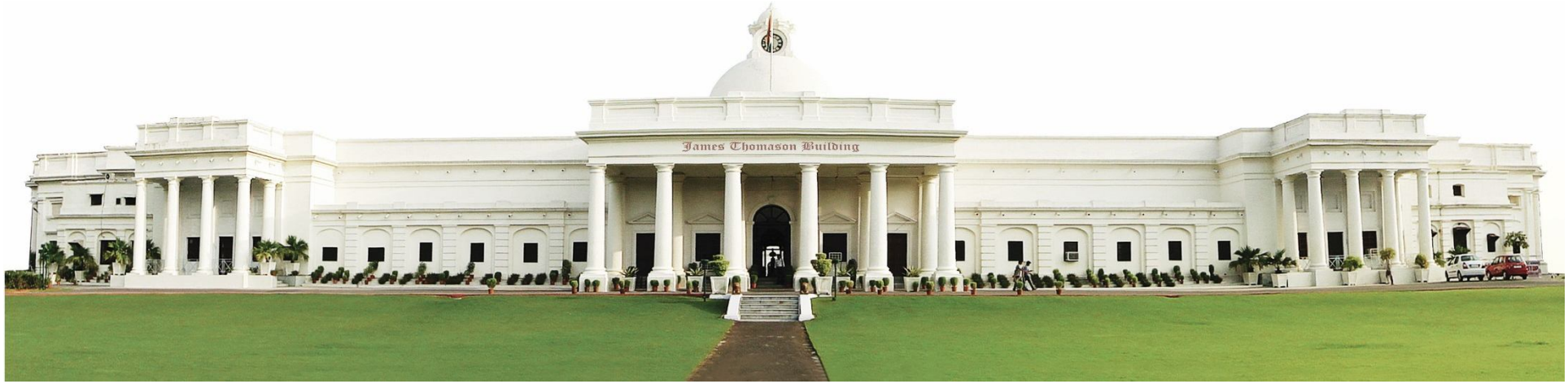
Department of Computer Science and Engineering

Indian Institute of Technology Roorkee

Roorkee 247 667

balarfcs@iitr.ac.in

https://sites.google.com/site/balaiitr/



Arrays

```
1  class Testarray{
2  public static void main(String args[]){
3
4  int a[]=new int[5];//declaration and instantiation
5  a[0]=10;//initialization
6  a[1]=20;
7  a[2]=70;
8  a[3]=40;
9  a[4]=50;
10
11 //printing array
12 for(int i=0;i<a.length;i++)//length is the property of array
13 System.out.println(a[i]);
14
15 }}
```

Terminal

```
sh-4.3$ javac Testarray.java
sh-4.3$ java Testarray
10
20
70
40
50
sh-4.3$
```

```java
class Testarray{
public static void main(String args[]){

int[] a;
a= new int [5];
a[0]=100;//initialization
a[1]=200;
a[2]=700;
a[3]=400;
a[4]=500;

//printing array
for(int i=0;i<a.length;i++)//length is the property of array
System.out.println(a[i]);

}}
```

Terminal

```
sh-4.3$ javac Testarray.java
sh-4.3$ java Testarray
100
200
700
400
500
sh-4.3$
```

Random Number Generation

```
1  import java.util.Random;
2  class TestarrayRan{
3  public static void main(String args[]){
4
5  int[] a;
6  a= new int [5];
7  Random rn = new Random();
8  for(int i =0; i < 5; i++)
9  {
10       a[i] = rn.nextInt(100) + 1;
11      System.out.println(a[i]);
12 }
13 }
14 }
```

Terminal

```
sh-4.3$ javac TestarrayRan.java
sh-4.3$ java TestarrayRan
92
56
58
69
62
sh-4.3$
```

WAP to find minimum of 5 numbers using array

```
1  import java.util.Random;
2  class TestarrayRan{
3  public static void main(String args[]){
4
5  int[] a;
6  a= new int [5];
7  int min;
8  Random rn = new Random();
9  for(int i =0; i < 5; i++)
10 {
11       a[i] = rn.nextInt(1000) + 1;
12       System.out.println(a[i]);
13 }
14 min=a[0];
15 for (int i=0; i<5; i++)
16 if (a[i]<min)
17 min=a[i];
18 System.out.println(min);
19 }
20 }
```

10 20 5 30 12

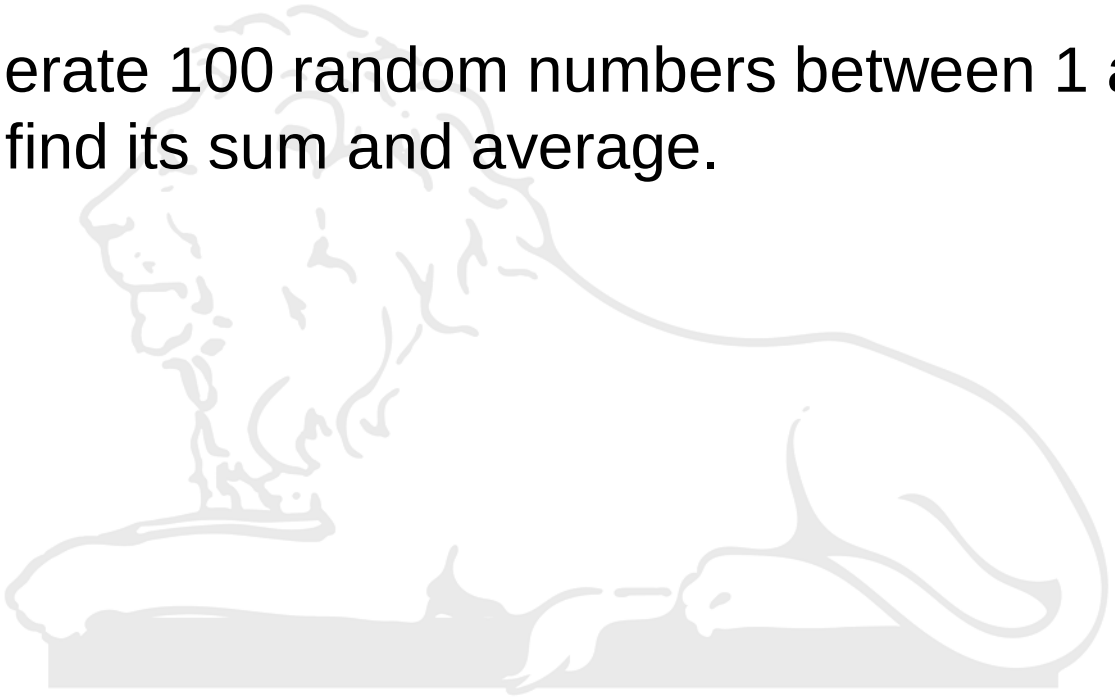
min a[1] a[2]

min

Terminal

```
sh-4.3$ javac TestarrayRan.java
sh-4.3$ java TestarrayRan
795
889
780
637
196
196
sh-4.3$
```

WAP to generate 100 random numbers between 1 and 1000 and then find its sum and average.



```java
import java.util.Random;
class SumAverage{
public static void main(String args[]){

int[] a;
a= new int [100];
int sum=0;
float average;
Random rn = new Random();
for(int i =0; i < 100; i++)
{
     a[i] = rn.nextInt(1000) + 1;
     sum=sum+a[i];
}
System.out.println("Sum of 100 number is =  " +sum);
average=(float) sum/(float) 100;
System.out.println("Average of 100 number is =  " +average);
}
}
```

Terminal

```
sh-4.3$ javac SumAverage.java
sh-4.3$ java SumAverage
Sum of 100 number is =  51658
Average of 100 number is =  516.58
sh-4.3$
```

```java
import java.util.Random;
import java.util.Scanner;
class SumAverage{
public static void main(String args[]){

int[] a;
a= new int [5];
int sum=0;
float average;
Random rn = new Random();
Scanner in = new Scanner(System.in);
for(int i =0; i < 5; i++)
{
     //a[i] = rn.nextInt(1000) + 1;
     a[i] = in.nextInt();
     sum=sum+a[i];
}
System.out.println("Sum of 5 number is =  " +sum);
average=(float) sum/(float) 5;
System.out.println("Average of 5 number is =  " +average);
}
}
```

Terminal

```
sh-4.3$ javac SumAverage.java
sh-4.3$ java SumAverage
10
20
30
40
50
Sum of 5 number is =  150
Average of 5 number is =  30.0
sh-4.3$
```

Identity Hash map of an Array

- Printing the Identity Hash map

Identity Hash map of an Array

```
1  class IdHashCode{
2  public static void main(String args[]){
3
4  int[] a;
5  a= new int [100];
6
7  System.out.println(a);
8  //System.out.println(a[1]);
9  }
10 }
11
```

Terminal

```
sh-4.3$ javac IdHashCode.java
sh-4.3$ java IdHashCode
[I@659e0bfd
sh-4.3$
```

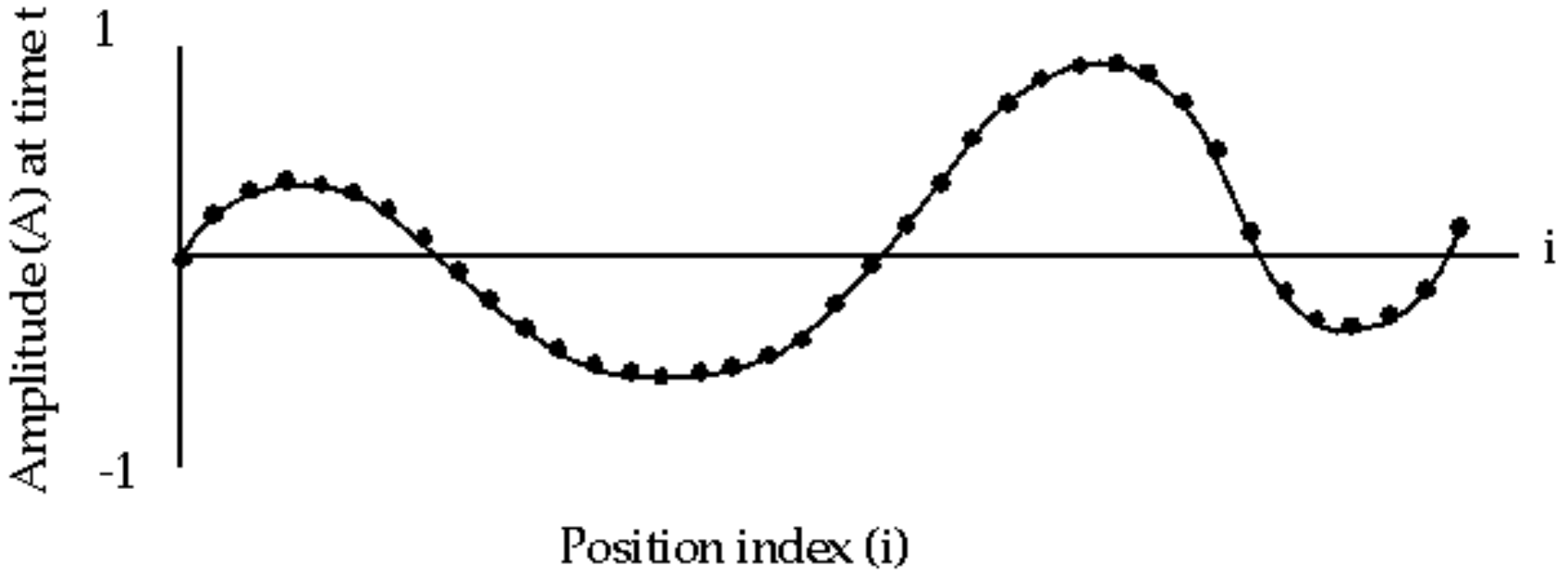
Length of an array

```
1  class ArrayLength{
2  public static void main(String args[]){
3
4  int a []= {10,20,30,40,50,60,70};
5  int n=a.length;
6
7  System.out.println(n);
8  }
9  }
```

Terminal

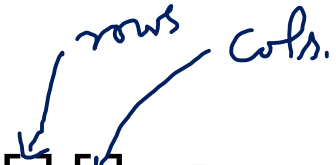
```
sh-4.3$ javac ArrayLength.java
sh-4.3$ java ArrayLength
7
sh-4.3$
```

Applications in signal Processing



Multidimensional array (2-D) in java

- Syntax



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

- Example
  - `int[][] arr=new int[3][3];//3 row and 3 column`

2-D Arrays

```
1  class Testarray3{
2  public static void main(String args[]){
3
4  //declaring and initializing 2D array
5  int arr[][]={{1,2,3},{2,4,5},{4,4,5}};
6
7  //printing 2D array
8  for(int i=0;i<3;i++){
9   for(int j=0;j<3;j++){
10    System.out.print(arr[i][j]+" ");
11   }
12   System.out.println();
13  }
14
15  }}
```

Terminal

```
sh-4.3$ javac Testarray3.java
sh-4.3$ java Testarray3
1 2 3
2 4 5
4 4 5
sh-4.3$
```

Left to Right

```
1  class Computef{
2  public static void main(String args[]){
3
4  int a,b,c,d,e,f;
5  a=10;
6  b=5;
7  c=20;
8  d=3;
9  e=2;
10 f=a&b<<d>>e;
11 System.out.println(f);
12 }
13 }
14
```

Terminal

```
sh-4.3$ javac Computef.java
sh-4.3$ java Computef
10
sh-4.3$
```

Right to Left

```
1  class Computef{
2  public static void main(String args[]){
3
4  int a,b,c;
5  a=7;
6  b=5;
7  c=10;
8  c+=b+=a;
9  System.out.println(a);
10 System.out.println(b);
11 System.out.println(c);
12 }
13 }
14
```

Terminal

```
sh-4.3$ javac Computef.java
sh-4.3$ java Computef
7
12
22
sh-4.3$
```