# **P1: PROGRAMS BASED ON ARRAYS BASICS IN JAVA[NCU]**

|  |
| --- |
| **Objective(s):**    To familiarize the students with linear data structure array and its basic operations |
| **Outcome:**    The students will be able to implement and use arrays for solving various problems |
| **Problem Statement:**    Create an array of integer with size n. Return the difference between the largest and the smallest value inside that array. |
| **Background Study:**    An Array is a data structure consisting of a collection of elements (values or variables), each identified by at least one array index or key. An array is stored such that the position of each element can be computed from its index tuple by a mathematical formula. The simplest type of data structure is a linear array, also called one-dimensional array. |
|  |
| **Code (Student Work Area):**    import java.util.Scanner;  public class Array\_Difference\_Largest\_Smallest{  public static void main(String[]args) {  Scanner sc= new Scanner(System.in);  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Array Making\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  System.out.println("Enter no. of Elements =");  int n = sc.nextInt();  int arr[]=new int[n];  for(int i=0;i<n;i++) {  System.out.println("Enter element");  arr[i]= sc.nextInt();    }  System.out.println("Elements in the array are=");  for(int i=0;i<n;i++) {  System.out.println( arr[i]);  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Largest and Smallest\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_    int large,small;  large=small=arr[0];  for (int i=1;i<n;i++) {  if(arr[i]> large){  large=arr[i];    }  if(arr[i]<small) {  small=arr[i];  }  }  System.out.println(large+"is the Largest No.");  System.out.println( small+"is the Smallest No. ");    //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Difference\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  int a =large-small;  System.out.println("Diiference is ="+ a);  }  }            **Output:** |
| **Question Bank:**   1. What is Data Structure?   Ans: *A data structure is a storage that is used to store and organize data. It is a way of arranging data on a computer so that it can be accessed and updated efficiently.*A data structure is not only used for organizing the data. It is also used for processing, retrieving, and storing data.   1. Why Array is called as Linear Data Structure?   Ans: A Linear data structure has data elements arranged in sequential manner and each member element is connected to its previous and next element. This connection helps to traverse a linear data structure in a single level and in a single run. Such data structures are easy to implement as computer memory is also sequential.An array is known as a linear data structure as it collects elements of the same data type and stores them in contiguous and adjacent memory locations.   1. What type of Indexing is used in Java?   Ans: **Indexing** is a data structure technique which allows us to quickly retrieve records from a database file. An Index is a small table having only two columns. The first column comprises a copy of the primary or candidate key of a table. Its second column contains a set of [pointers](https://www.guru99.com/c-pointers.html) for holding the address of the disk block where that specific key value is stored.   * Takes a search key as input * Efficiently returns a collection of matching records.   Indexing in Database is defined based on its indexing attributes. Two main types of indexing methods are:   * Primary Indexing * Secondary Indexing   **Primary Index in DBMS**  Primary Index is an ordered file which is fixed length size with two fields. The first field is the same as a primary key and second, filed is pointed to that specific data block. In the primary Index, there is always one to one relationship between the entries in the index table.  The primary Indexing in DBMS is also further divided into two types.   * Dense Index * Sparse Index   **Dense Index**  In a dense index, a record is created for every search key valued in the database. This helps you to search faster but needs more space to store index records. In this Indexing, method records contain search key value and points to the real record on the disk.  **Sparse Index**  It is an index record that appears for only some of the values in the file. Sparse Index helps you to resolve the issues of dense Indexing in DBMS. In this method of indexing technique, a range of index columns stores the same data block address, and when data needs to be retrieved, the block address will be fetched.However, sparse Index stores index records for only some search-key values. It needs less space, less maintenance overhead for insertion, and deletions but It is slower compared to the dense Index for locating records.  **Secondary Index in DBMS**  The secondary Index in DBMS can be generated by a field which has a unique value for each record, and it should be a candidate key. It is also known as a non-clustering index.This two-level database indexing technique is used to reduce the mapping size of the first level. For the first level, a large range of numbers is selected because of this; the mapping size always remains small.   1. How to find the missing number in integer array of 1 to 100?   Ans: import java.util.Scanner;  public class Array\_Missing\_Number {  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  System.out.println("Enter the n value: ");  int n = sc.nextInt();  int inpuArray[] = new int[n];  System.out.println("Enter (n-1) numbers: ");  for(int i=0; i<=n-2; i++) {  inpuArray[i] = sc.nextInt();  }  //Finding the missing number  int sumOfAll = (n\*(n+1))/2;  int sumOfArray = 0;  for(int i=0; i<=n-2; i++) {  sumOfArray = sumOfArray+inpuArray[i];  }  int missingNumber = sumOfAll-sumOfArray;  System.out.println("Missing number is: "+missingNumber);  }  }   1. How to find the second-highest value in a numeric array.   Ans:  public class Array\_Second\_Largest{  public static int getSecondLargest(int[] a, int total){  int temp;  for (int i = 0; i < total; i++)  {  for (int j = i + 1; j < total; j++)  {  if (a[i] > a[j])  {  temp = a[i];  a[i] = a[j];  a[j] = temp;  }  }  }  return a[total-2];  }  public static void main(String args[]){  int a[]={1,2,5,6,3,2};  int b[]={44,66,99,77,33,22,55};  System.out.println("Second Largest: "+getSecondLargest(a,6));  System.out.println("Second Largest: "+getSecondLargest(b,7));  }}   1. How to swap the first and last elements of an array.   Ans:import java.util.Arrays;    public class Array\_Swap\_First\_Last{    public static void main(String args[]) {  int[] arr = {10, 20, 30, 40};  System.out.println(Arrays.toString(arr));  swapElements(arr);  System.out.println(Arrays.toString(arr));  }    public static void swapElements(int[] arr) {  int tmp = arr[0];  arr[0] = arr[arr.length - 1];  arr[arr.length - 1] = tmp;  }  }  1. Write a Java Program to check if see if Array contains a specific value. (Linear Search)  Ans:  import java.util.Scanner;  public class Array\_Searching{  public static void main(String[]args) {  Scanner sc= new Scanner(System.in);  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Array Making\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  System.out.println("Enter no. of Elements =");  int n = sc.nextInt();  int arr[]=new int[n];  for(int i=0;i<n;i++) {  System.out.println("Enter element");  arr[i]= sc.nextInt();      }  System.out.println("Elements in the array are=");  for(int i=0;i<n;i++) {  System.out.println( arr[i]);  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Searching\_\_\_\_\_\_\_\_\_\_\_\_\_  System.out.println("--SEARCHING--");  System.out.println("Element to be found");//Searching  int flag=0;  int ele= sc.nextInt();  for(int i=0;i<n;i++) {  if(arr[i]==ele) {  flag++;  System.out.println("element found! at index" + i);  break;  }    }  if(flag==0) {  System.out.println("Not Found");    }  }  } |