# Arrays

1. Differentiate between searching and sorting.

   (2018)

   Answer:
   Searching is a technique which is used to search a particular element in an array or string. Sorting is a technique which is used to rearrange the elements of an array or string in a particular order either ascending or descending.

2. What is the difference between the linear search and the binary search technique?

   (2019)

   Answer :

| **Linear search** | **Binary search** |
|---|---|
| Linear search works both for sorted and unsorted data. | Binary search works on sorted data (either in ascending order or in descending order). |
| Linear search begins at the start of an array i.e. at 0th position. | This technique divides the array in two halves, and the desired data item is searched in the halves. |

3. Consider the following String array and give the output

   ```
   String arr[]= {“DELHI”, “CHENNAI”, “MUMBAI”, “LUCKNOW”, “JAIPUR”};
   System.out.println(arr[0] .length( )> arr[3] .length( );
   System.out.print(arr[4] ,substring(0,3));
   ```

   (2018)

   Answer:
   false (at index 0, DELHI consists of 5 characters, at index 3, LUCKNOW consists of 7 characters. Therefore 5 > 7 is false)
   JAI (at index 4, JAIPUR exists and extract its three characters)

4. Write a program to input 15 integer elements in an array and sort them in ascending order using the bubble sort technique.

(2019)

Answer :

```
import java.io.*;
import java.util'*;
class AscendingOrder
{ public static void main(String args[])
   { int i, j, temp;
     Scanner sc = new Scanner(System.in);
     int arr[] = new int[15];
     System.out.println("Enter 15 integers:");
     for (i=0 ; i<=15 ; i++)
        { arr[i] = sc.nextlnt( );
          for(i=0 ; i<14 ; i++)
            { for(j=0 ; j<14-i ; j++)
                 { if(arr[j] > arr[j+1])
                    { temp = arr[j];
                      arr[j] = arr [j + 1];
                      arr[j + 1] = temp;
                     }
                 }
            }
          System.out.println("Elements in ascending order are::");
          for (i = 0; i < 15; i+ +)
            { System.out.println(arr[i]);
            }
        }
    }
}
```

5. Write a program to accept the name and total marks of N number of students in two single subscript array name[] and totalmarks[ ].
   Calculate and print:
   (i) The average of the total marks obtained by N Number of students.
   [average = (sum of total marks of all the students)/N]
   (ii) Deviation of each student's total marks with the average.
   [deviation = total marks of a student – average] '

   (2018)

   Answer:

```
import java.io.*;
```

```
import java. util. Scanner;
class NameMarks
{ public static void main(String argsO) throws IOException
   { Scanner sc = new Scanner(System.in);
     System.out.print("Enter number of students:");
     int N = sc.nextInt( );
     String named = new String[N];
     int totalmarksG = new int[N];
     double deviation[ ] = new double[N];
     double sum = 0;
     for (int i=0 ; i<N ; i++)
        { System.out.print("Enter Name of the Student:");
          name[i] = sc.next( );
          System.out.print("Enter Marks:");
          totalmarks[i] = sc.nextIntO;
          sum = sum + totalmarks [i];
        }
     double average = sum / N;
     System.out.println("The average of the total marks of " +N+" number of
     students:" +average);
     for (int i=0 ; i<N ; i++)
        { deviadon[i] = total marks (i] – average;
          System.out.println("Deviation of" + name[i] + "s marks with the average:"
          +deviation[i]);
        }
    }
}
```

6. If int x [ ] = { 4, 3,7, 8, 9,10}; what are the values of p and q ?
   (i) p = x.length
   (ii) q = x[2] + x[5] * x[1]

   Answer:
   (i) 6
   (ii) 37
   Method: q = x[2] + x[5] * x[1]
   = 7 + 10 * 3
   = 7 + 30
   = 37

7. Write a program to input integer elements into an array of size 20 and perform the following operations:
(i) Display largest number from the array.
(ii) Display smallest number from the array.
(iii) Display sum of all the elements of the array.

(2017)

Answer:

```
import java.util.Scanner;
class LargeSmallSum
{ public static void main(String args[ ])
   { int n;
     int max, min, sum = 0;
     int i, j;
     Scanner s = new Scanner(System.in);
     System.out.print("Enter no. of elements you want in array:");
     n = s.nextInt();
     int a[ ] = new int[n];
     System.out.println("Enter all the elements:");
     for (i=0 ; i<n ; i++)
        { a[i] = s.nextInt( );
       }
     max = a[0];
     min = a[0];
     for (i=0 ; i<n ; i++)
        { if(a[i] > max)
            { max = a[i];
             }
           if (a[i] < min)
             { min = a[i];
            }
        }
     System.out.println("Maximum Number is:"+max);
     System.out.println("Smallest Number is:" +min);
     for(i=0 ; i<n ; i++)
        { sum = sum + a[i];
        }
     System.out.println("Sum of the Numbers is:" +sum);
```

```
    }
}
```

8. Write a program to input forty words in an array. Arrange these words in descending order of alphabets, using selection sort technique. Print the sorted array.

(2017)

Answer:

```
import java.util.Scanner;
class Descending
{ public static void main(String args[ ])
   {int n;
    String temp;
    Scanner s = new Scanner(System.in);
    System.out.print("Enter number of words you want to enter:");
    n = s.nextInt( );
    String names[ ] = new String[n];
    System.out.println("Enter all words:");
    for(int i=0 ; i<n ; i++)
      { names[i] = sc..nextLine( );
     }
    for(int i=0 ; i<n ; i++)
      { for (int j=i+1 ; j<n ; j++)
           { if (names[i].compareTo(names[i+1]) < 0)
               {temp = names [i];
                names[i] = names[i+1];
                names[i+1] = temp;
               }
           }
      }
    System.out.print("Words in Descending Order:");
    for (int i=0 ; i<n-l ; i++)
       {System.out.print(names[i] + ",");
       }
    System.out.print(names[n – 1]);
    } }
```

9. Write a program to initialise the seven Wonders of the World along with their locations in two different arrays. Search for a name of the country input by the user. If found, display the name of the country along with its Wonder, otherwise display "Sorry Not Found!".
**Seven wonders** — CHICHEN ITZA, CHRIST THE REDEEMER, TAJMAHAL, GREAT WALL OF CHINA, MACHU PICCHU, PETRA, COLOSSEUM
**Locations** — MEXICO, BRAZIL, INDIA, CHINA, PERU, JORDAN, ITALY
**Example** — Country Name: INDIA Output: INDIA-TAJMAHAL
Country Name: USA Output: Sorry Not Found!

(2016)

Answer:

```
import java.util.Scanner;
class test
{
    public static void main( )
    {
        String w[ ] = {"Chichen Itza", "Christ The Redeemer", "Taj Mahal",
                       "Great wall of China","Machu Picchu"; "Petra Colosseum" };
        String L [ ] = {"Mexico, "Brazil", "India", "China", "Peru", Jordan",
                        "Italy"};
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter country Name");
        String c=sc.next();
        boolean found = false;
        for (int i=0; i<w.length( ); i++)
        {
            if (L [i].equalsIgnoreCase(c))
            {
                found = true;
                System.out.println (L[i] + "–" + w[i]);
                break;
            }
        }
            if (found = = false)
            System.out.println ("Sorry Not found");
        }
        }
```

10. Find the errors in the given program segment and re-write the statements correctly to assign values to an integer array.

```
int a = new int (5);
for (int i=0 ; i<=5 ; i++) a [i] = i;
```

(2016)

Answer:
The corrected code is

```
int [ ] a = new int [5];
for(int i=0 ; i<=4 ; i++)
  {a[i]=i;
 }
```

11. What will this code print?

```
int arr[]=new int[5];
System.out.println(arr);
```

(i) 0. (ii) value stored in arr[0]. (iii) 0000. (iv) garbage value

(2015)

Answer:
(iv) garbage value

12. Write a program to input and store roll numbers, names and marks in 3 subjects of n number students in five single-dimensional arrays and display the remark based on average marks as given below : (The maximum marks in the subject are 100)

$$Average\ marks = \frac{Total\ Marks}{3}$$

| *Average marks* | *Remark* |
|---|---|
| 85—100 | EXCELLENT |
| 75—84 | DISTINCTION |
| 60—74 | FIRST CLASS |
| 40—59 | PASS |
| Less than 40 | POOR |

Answer:

(2015)

```
import java.util.*;
class test
{
    Public static void main ( )
    {
        Scanner S = new scanner (system.in);
        int n;
        System.out.println ("Enter no. of student");
        n = S.nextInt ( );
        int R[ ] = new int [n];
        string nm [ ] = new string [n];
        int n1[ ] = new int[n];
        int n2[ ] = new int[n];
        int n3[ ] = new int[n];
        for (int k = 0; k < n; k++)
    {
        System.out.println ("Enter Roll no");
        R[k] = S.nextInt ( );
        System.out.println ("Enter Name");
        nm[k] = S.nextLine ( );
        System.out.println ("Enter marks in 3 subjects");
        n1[k] = S.nextInt ( );
      n2[k] = S.nextInt ( );
      n3[k] = S.nextInt ( );-
}
      double avg; string remark;
for (int k=0; k < n; k++)
{
      avg = (n1[k] + n2[k] + n3[k])/3.0;
      if (avg < 40)
          Remark = "Poor";
      else if (avg > = 40 && avg < = 59)
          Remark = "Pass";
      else if (avg > = 60 && avg < = 74)
          Remark = "FIRST CLASS";
      else if (avg > = 75 && avg < = 84)
          Remark = "DISTINCTION";
      else
          Remark = "EXCELLENT";
      System.out.println [nm[k] + "\t" + avg + "\t" + Remark);
}
}
}
```

13. Write a program to input twenty names in an array. Arrange these names in descending order of alphabets, using the bubble sort technique.

(2015)

Answer:

```
import java.util.scanner;
class sort
{
      public static void main ( )
      {
            String nm[ ] = new string [20];
            Scanner S = new Scanner (System.in);
            for (int i=0; i<=19; i++)
      {
            System.out.println ("Enter Name");
            nm[i]=SnextLine( );
      }
      String t;
      for (int i=0; i<=18; i++)
      {
            for (int j=0; j<=18–i; j++)
      {
            if (nm[j].compareTo(nm [j+1])<0)
            {
                t=nm[i];
                nm[i]=nm[j];
                nm[j]=t;
            }
      }
}
for (int i=0; i<=19; i++)
      {
            System.out.println (nm [i]);
      }
      }
}
```

14. If int n [ ] = {1, 2, 3, 5, 7, 9, 13, 16}, what are the values of x and y ?

    x = Math.pow (n[4], n [2]);

    y = Math.sqrt (n[5]+n[7];

(2013)

Answer:

x = 343, y = 5

15. Write a program to input 10 integer elements in an array and sort them in descending order using the bubble sort technique.

(2013)

Answer:

```
public class Bubble sort
{
public static void main (string [ ] args)
{
int [ ] array =  {41, 15, 25, 77, 45, 37, 23, 70, 27, 9}
System.out.println ("Initially.array is : ");
for (int i = 0; i < array.length; ++ i)
System.out.print (" " + array [i]);
System.out.println ( );
bubble_sort (array);
System.out.println ("After Bubble sort, array indescending order is : ");
for (int i = 0; i = array.length ; ++ i);
System.out.print (" "+ array [i]);
System.out.println ( );
}
public static void bubble_sort (int [ ] num)
{
int j;
int temp; // holding variable
for (int i = 0; i < num.length –1; i++)
{
for (j = 0; j < num.length – i–1; j++)
{
if (num [j] < num [j+1])          // Change to > for ascending sort
{
temp = num [j];                   // Swap elements
num [j] = num. [j+1];
num [j+1] = temp;
}
}
}
}
}
```

16. State the total size in bytes, of the arrays a[4] of char data type and p[4] of float data type.

(2011)

Answer:

Size of char a[4] is 4 × 2 = 8 Bytes.

Size of float p[4] is 4 × 4 = 16 Bytes.

17. What is an array? Write a statement to declare an integer array of 10 elements.

(2012)

Answer:

An array is a group of similar type variables that are referenced by a common name.

int arr = new int [10];

18. Name the search or sort algorithm that :

(i) Makes several passes through the array, selecting the next smallest item in the array each time and placing it where it belongs in the array.

(ii) At each stage, compare the sought key value with the key value of the middle element of the array.

(2012)

Answer:

(i) Selection sort

(ii) Binary search

19. Write one difference between Linear Search and Binary Search.

(2011)

Answer:

| Linear search | Binary search |
|---|---|
| Linear search can be performed on unsorted arrays. | Binary search can be performed on only sorted arrays. |

20. Write a program to input and store the weight of ten people. Sort and display them in descending order using the selection sort technique.

Answer:

(2011)

```
import java.io.*;
public class sortData
{
    public static void main ( ) throws IOException
    {
        double wt [ ] = new double [10];
        BufferedReader br=new BufferedReader
                                (new InputStreamReader (System.in));
        System.out.println("Enter weight of 10 People");
        for (int i = 0; i < 10; i ++)
        {
            wt[i] = Double.parseDouble (br.readLine ( ));
        }
        int j; double t;
        for (j = 0; j < = 8; j ++)
        {
            for (int k = j+1; k < = 9; k ++)
            {
                if (wt[j] < wt [k])
                {
                    t = wt [j];
                    wt [j] = wt [k];
                    wt [k] = t;
                }
            }

        }
    for (int R = 0; R < = 9;  R ++)
    {
        System.out.println(wt [R]);
    }
  }
}
```

21. Write a program to accept the names of 10 cities in a single dimension string array and their STD (Subscribers Trunk Dialling) codes in another single dimension integer array. Search for a name of a city input by the user in the list. If found, display* "Search successful" and print the name of the city along with its STD code, or else display the message "Search Unsuccessful, No such city in the list".

    Answer:

(2012)

```
Class cities
{void main (String city_nam)
{String City [ ]={"Agra", "Delhi", "Kanpur", "Bombay", "Lucknow", "Allahabad",
"Meerut", "Varanasi", "Mathura", "Aligarh"};
int code [ ] = {0562, 0001, 0555, 0123, 0257, 0565, 0375, 0999, 0552, 0543};
int i, c=0;
for (i=0; i<=10; i++)
{if (city_nam.equalsIgnore Case(City[i]))
{C++;
System.out.println("search successful");
System.out.println(city [i] + " " + code [i]);
}
}
if (C==0)
System.out.println("search unsuccessful, No such city in the list");
} }
```

22. Write a program to store 6 elements in any array P, and 4 elements in an array Q and produce a third array R, containing all the elements of array P and Q. Display the resultant array.

| EXAMPLE | INPUT | OUTPUT |
|---|---|---|
| P[ ] | Q[ ] | R[ ] |
| 4 | 19 | 4 |
| 6 | 23 | 6 |
| 1 | 7 | 1 |
| 2 | 8 | 2 |
| 3 | | 3 |
| 10 | | 10 |
| | | 19 |
| | | 23 |
| | | 7 |
| | | 8 |

(2010)

Answer:

```
import java.io.*;
class joinarry
{
	public static void main()throws IOException
	{
		InputStreamReader IR=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader (IR);
		int P[ ]= new int[6];
		int Q[ ]=new int[4];
		int R[ ]=new int[10];
		for(int i=0;i<6;i++)
		{
			System.out.println("Enter the number :");
			P[i]=Integer.parseInt(br.readLine());
		}
		for (i=0;i<4;i++)
		{
			System.out.println("Enter the number :");
			Q[i]=Integer.parseInt(br.readLine());
		}
		for (i=0; i<6; i ++)
		{
			R[i] = P [i];
		}
		for (i=6; i<10; i ++)
		{

		    R[i]=Q[i–6];
		}
	System.out.println("P[ ]\tQ[ ]\t\tR[ ]");
	for(i=0;i<10;i++)
	{
		if(i<4)
		{
			System.out.println(P[i]+"\t"+Q[i]+"\t\t"+R[i]);
		}
		else if(i<6)
		{
			System.out.println(P[i]+"\t\t\t"+R[i]);
		}
		else
		{
			System.out.println("\t\t\t"+R[i]); //space printing instead of array
		}
	   }
	}
}
```

23. Write a program to perform binary search on a list of integers given below, to search for an element input by the user, if it is found display the element along with its position, otherwise, display the message "Search element not found".
5, 7, 9, 11, 15, 20, 30, 45, 89, 97

(2010)

Answer:

```
import java.io.*;
class binary
{
    public static void main() throws IOException
    {
        InputStreamReader IR=new InputStreamReader(System.in);
        BufferedReader br=new BufferedReader(IR);
        int arr[]={5, 7, 9, 11, 15, 20, 30, 45, 89, 97   //static input
        int beg, mid, end, pos;
        System.out.print("Enter the Number to be searched");
        x=Integer.parseInt(be.readLine());
        beg = 0; end = a.length –1;
        pos = –1;
        while(beg <=end && pos = = –1)
        {
            mid=(beg+end)/2;
            if(arr[mid]==x)
            {
                pos = mid;
            }
            else if(x > arr[mid])
            {
                beg = mid+1;
            }
            else
            {
                end=mid–1;
            }
        }
        if(pos != –1)
        {
            System.out.println(arr[x]+ "was found at position" +mid);
        }
        else
        {
            System.out.println("Search element not found");
        }
    }           //end of main
}               //end of class
```

24. Write a program to accept the year of graduation from school as an integer value from the user. Using the Binary Search technique on the sorted array of integers given below.
Output the message “Record exists” If the value input is located in the array. If not, output the message “Record does not exist”.
{1982, 1987, 1993, 1996. 1999, 2003, 2006, 2007, 2009, 2010}

(2014)

Answer:

```
import java.io.*;
class Search
{
        public static void main ( ) throws IOException
        {
                int [ ] year={1982,1987,1993,1996,1999,2003,2006,2007,2009,2010};
                InputStreamReader IR = new InputStreamReader (System.in);
                BufferedReader br=new BufferedReader (IR);
                System.out.println("Enter year of Graduation :");
                int y = Integer.parseInt (br. readLine ( ));
                int beg, mid, end, pos;
                beg = 0;
                end = 9;
                pos = – 1;
                while (beg < = end && pos = = – 1)
                {
                    mid = (beg + end)/2;
                    if (y = = year [mid])
                    pos = mid;
                else
                    if (y > year [mid])
                    beg = mid + 1;
                else
                    end = mid – 1;
                }
                if (pos ! = – 1)
                System.out.println ("Record exists");
                else
                System.out.println ("Record does not exist");
        }
}
```