**Assignment 4:**

1. **Implement Bubble Sort for Application.**

**Source Code:**

**import java.util.Scanner;**

**public class BubbleSort {**

**private int arr[];**

**private int size;**

**BubbleSort(int size) {**

**this.size = size;**

**arr = new int[size];**

**}**

**public void bubbleSort(int arr[], int n) {**

**for (int i = 0; i < n; i++) {**

**for (int j = 0; j < n - 1; j++) {**

**if (arr[j] > arr[j + 1]) {**

**int temp = arr[j];**

**arr[j] = arr[j + 1];**

**arr[j + 1] = temp;**

**}**

**}**

**}**

**}**

**public static void main(String[] args) {**

**Scanner sn = new Scanner(System.in);**

**int n = sn.nextInt();**

**BubbleSort bs = new BubbleSort(n);**

**int arr[] = new int[n];**

**for (int i = 0; i < n; i++)**

**arr[i] = sn.nextInt();**

**sn.close();**

**System.out.println("Before Sorting");**

**for (int i = 0; i < n; i++)**

**System.out.print(arr[i] + " ");**

**System.out.println();**

**System.out.println("After Sorting");**

**bs.bubbleSort(arr, n);**

**for (int i = 0; i < n; i++)**

**System.out.print(arr[i] + " ");**

**}**

**}**

**Output:**

4

5 3 2 9

**Before** Sorting

5 3 2 9

After Sorting

2 3 5 9

1. **Input a user defined array and sort it using bubble sort and search and element using binary search.**

**Source Code:**

**import java.util.Scanner;**

**public class BinarySearch {**

**private int arr[];**

**private int size;**

**private int key;**

**BinarySearch(int size, int key) {**

**this.size = size;**

**this.key = key;**

**arr = new int[size];**

**}**

**public boolean binarySearch(int arr[], int n, int key) {**

**int low = 0, high = n - 1;**

**while (low <= high) {**

**int mid = (low + high) / 2;**

**if (arr[mid] == key)**

**return true;**

**else if (arr[mid] < key)**

**low = mid + 1;**

**else**

**high = mid - 1;**

**}**

**return false;**

**}**

**public static void main(String[] args) {**

**Scanner sn = new Scanner(System.in);**

**int n = sn.nextInt();**

**int key = sn.nextInt();**

**BinarySearch b = new BinarySearch(n, key);**

**BubbleSort bs = new BubbleSort(n);**

**int arr[] = new int[n];**

**for (int i = 0; i < n; i++)**

**arr[i] = sn.nextInt();**

**sn.close();**

**bs.bubbleSort(arr, n);**

**System.out.println("Performing binary search on array for key: " + key);**

**if (b.binarySearch(arr, n, key))**

**System.out.println("Key Found");**

**else**

**System.out.println("Not Found");**

**}**

**Output:**

4

3

9 5 3 2

**Performing** binary **search** on array for key: 3

**Key** Found

1. **Write a menu driver program to find the item from string array / integer array based on the user input using binary search.**

**import java.util.Scanner;**

**import java.util.Arrays;**

**public class SearchIntegerString {**

**private int arr[];**

**private String names[];**

**private int size;**

**private int key;**

**String name;**

**SearchIntegerString(int size, int key, String name)**

**{**

**this.size = size;**

**this.key = key;**

**this.name = name;**

**arr = new int[size];**

**names = new String[size];**

**}**

**public boolean binarySearchInt(int arr[], int n, int key)**

**{**

**int low = 0, high = n-1;**

**while(low <= high)**

**{**

**int mid = (low+high)/2;**

**if(arr[mid] == key)**

**return true;**

**else if(arr[mid] < key)**

**low = mid+1;**

**else**

**high = mid-1;**

**}**

**return false;**

**}**

**public boolean binarySearchString(String[] names, int n, String name) {**

**int low = 0, high = n-1;**

**while(low <= high)**

**{**

**int mid = (low+high)/2;**

**if(names[mid].equals(name))**

**return true;**

**else if(arr[mid] < key)**

**low = mid+1;**

**else**

**high = mid-1;**

**}**

**return false;**

**}**

**public static void main(String[] args)**

**{**

**Scanner sn = new Scanner(System.in);**

**System.out.println("Enter Size of Array");**

**int n = sn.nextInt();**

**int arr[] = new int[n];**

**String names[] = new String[n];**

**System.out.println("Enter Integer Elements");**

**for(int i = 0; i < n; i++)**

**arr[i] = sn.nextInt();**

**System.out.println("Enter String Elements");**

**for(int i = 0; i < n; i++)**

**names[i] = sn.next();**

**System.out.println("Enter Integer to be Searched");**

**int key = sn.nextInt();sn.nextLine();**

**System.out.println("Enter String to be Searched");**

**String name = sn.nextLine();**

**SearchIntegerString s = new SearchIntegerString(n, key, name);**

**System.out.println("Enter Choice:\n1: To Find Integer Element \n2: To Find String \n3: To Exit");**

**int choice = sn.nextInt();**

**switch (choice) {**

**case 1:**

**Arrays.sort(arr);**

**System.out.println("Performing binary search on array for key: "+key);**

**if(s.binarySearchInt(arr, n, key))**

**System.out.println("Key Found");**

**else**

**System.out.println("Not Found");**

**break;**

**case 2:**

**Arrays.sort(names);**

**System.out.println("Performing binary search on array for string: "+name);**

**if(s.binarySearchString(names, n, name))**

**System.out.println("Name Found");**

**else**

**System.out.println("Not Found");**

**break;**

**default:**

**break;**

**}**

**}**

**}**

**Output:**

Enter Size of Array

3

Enter Integer Elements

6 2 4

Enter String Elements

CC AA DD

Enter Integer to be Searched

4

Enter String to be Searched

DDD

Enter Choice:

1: To Find Integer Element

2: To Find String

3: To Exit

1

Performing binary search on array for key: 4

Key Found

1. **Insertion Sort.**

**import java.util.Scanner;**

**public class InsertionSort {**

**private int arr[];**

**private int size;**

**InsertionSort(int size) {**

**this.size = size;**

**arr = new int[size];**

**}**

**public void insertionSort(int arr[], int n) {**

**for (int i = 0; i < n; i++) {**

**int curr = arr[i];**

**int j = i - 1;**

**while (j >= 0 && arr[j] > curr) {**

**arr[j + 1] = arr[j];**

**j--;**

**}**

**arr[j + 1] = curr;**

**}**

**}**

**public static void main(String[] args) {**

**Scanner sn = new Scanner(System.in);**

**int n = sn.nextInt();**

**InsertionSort is = new InsertionSort(n);**

**int arr[] = new int[n];**

**for (int i = 0; i < n; i++)**

**arr[i] = sn.nextInt();**

**System.out.println("Before Sorting");**

**for (int i = 0; i < n; i++)**

**System.out.print(arr[i] + " ");**

**System.out.println();**

**System.out.println("After Sorting");**

**is.insertionSort(arr, n);**

**for (int i = 0; i < n; i++)**

**System.out.print(arr[i] + " ");**

**}**

**}**

**Output:**

4

5 6 3 8

**Before** Sorting

5 6 3 8

After Sorting

3 5 6 8