

|                |                           |
|----------------|---------------------------|
| <b>NAME</b>    | <b>MUHAMMAD ILYAS</b>     |
| <b>ID NO</b>   | <b>15392</b>              |
| <b>DEGREE</b>  | <b>BS (SOFTWARE ENGR)</b> |
| <b>LAB</b>     | <b>DATA STRUCTURE</b>     |
| <b>TEACHER</b> | <b>SIR-M.ADIL</b>         |

### **TASK#1;**

Design a linear array B [] of size 7 elements. Put the following elements in it. s, u, g, a, z, e, y a. Implement the Linear Search Algorithm on it to find "g" and display the message Element g is found successfully b. Search for element "m" and message should be displayed Search is Unsuccessful

### **PART# A-**

Implement the Linear Search Algorithm on it to find "g" and display the message Element g is found successfully.

```
package linearArray;

public class Main {
    public static void main(String[]args) {
        char A[]= {'s','u','g','a','z','e','y'};
        boolean found = false;
        int i =1;
        int max =6;
        char item='g';

        while(i<max && found==false) {
            if(A[i]==item)
            {
```

```

        found=true;
    }
    else
    {
        i++;
    }
}
if(found==true)
{
    System.out.println("element 'g' is found successfully"
);
}
else
{
    System.out.println("search is unsuccessful");
}

}
}

```

The screenshot shows an IDE with two windows. The left window displays the source code for 'Main.java', which is a Java program to search for the character 'g' in an array. The right window shows the console output, which is 'element 'g' is found successfully'.

```

Main.java
linearArray > src > linearArray > Main > main(String[]): void
4 public static void main(String[] args) {
5     char A[] = {'s', 'u', 'g', 'a', 'z', 'e', 'y'};
6     boolean found = false;
7     int i = 1;
8     int max = 6;
9     char item = 'g';
10
11     while(i < max && found == false) {
12         if(A[i] == item)
13         {
14             found = true;
15         }
16         else
17         {
18             i++;
19         }
20     }
21     if(found == true)
22     {
23         System.out.println("element 'g' is found successfully");
24     }
25     else
26     {
27         System.out.println("search is unsuccessful");
28     }
}

```

Problems | Javadoc | Declaration | Console

```

<terminated> Main (8) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 2, 2020, 6:04:43)
element 'g' is found successfully

```

## PART # B;

b. Search for element "m" and message should be displayed Search is Unsuccessful.

```
package linearArray2;

public class Main {
    public static void main(String[]args) {
        char A[]= {'s','u','g','a','z','e','y'};
        boolean found = false;
        int i =1;
        int max =6;
        char item='m';

        while(i<max && found==false) {
            if(A[i]==item)
            {
                found=true;
            }
            else
            {
                i++;
            }
        }
        if(found==true)
        {
            System.out.println("element 'g' is found
successfully" );
        }
        else
        {
            System.out.println("search is unsuccessful");
        }

    }
}
```

}



The screenshot shows an IDE with two windows. The left window displays the source code for a Java program named 'Main.java'. The code defines a character array 'A' with the characters 's', 'u', 'g', 'a', 'z', 'e', 'y'. It then searches for the character 'm' using a while loop. Since 'm' is not found, the program prints 'search is unsuccessful'. The right window is the console, which shows the output '<terminated> Main (9) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 2, 2020, 6:02:54 P) search is unsuccessful'.

```
5 char A[] = {'s', 'u', 'g', 'a', 'z', 'e', 'y'};
6 boolean found = false;
7 int i = 1;
8 int max = 6;
9 char item = 'm';
10
11 while(i < max && found == false) {
12     if(A[i] == item)
13     {
14         found = true;
15     }
16     else
17     {
18         i++;
19     }
20 }
21 if(found == true)
22 {
23     System.out.println("element 'g' is found successfully");
24 }
25 else
26 {
27     System.out.println("search is unsuccessful");
28 }
29
30
```

## TASK#2;

**Suppose there is a list of 6 unsorted elements. 15, 10, 12, 11, 9, 10 Design a Program to create an array A [] and store this list in it, and then apply Insertion Sort Algorithm to Sort the list**

```
public class Main {
    public static void main(String[] args) {
        int [] numbers = {15, 10, 12, 11, 9, 10};
        sort(numbers);
        printArray(numbers);
    }
}
```

```

public static int[] sort(int[] A) {
    for(int i=1; i<A.length; i++) {
        int key=A[i];
        int j=i-1;
        while(j>=0 && A[j]>key) {
            A[j+1]= A[j];
            j--;
        }
        A[j+1] =key;
    }

return A;

}

public static void printArray(int[] A) {
    for(int i =0; i<A.length; i++) {
        System.out.println(A[i]);
    }
}
}
}

```

The screenshot shows an IDE window with two panes. The left pane displays the source code for a Java application, and the right pane shows the console output.

**Source Code (Left Pane):**

```

Main.java
src > insertion > Main > sort(int[]): int[]
3 public class Main {
4     public static void main(String[] args) {
5         int [] numbers= {15,10,12,11,9,10};
6         sort(numbers);
7         printArray(numbers);
8     }
9 }
10 public static int[] sort(int[] A) {
11     for(int i=1; i<A.length; i++) {
12         int key=A[i];
13         int j=i-1;
14         while(j>=0 && A[j]>key) {
15             A[j+1]= A[j];
16             j--;
17         }
18         A[j+1] =key;
19     }
20 }
21 return A;
22 }
23 }
24 public static void printArray(int[] A) {
25     for(int i =0; i<A.length; i++) {
26         System.out.println(A[i]);
27     }
28 }

```

**Console Output (Right Pane):**

```

<terminated> Main (7) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 2, 2020, 4:32:45 PM)
9
10
10
11
12
15

```

