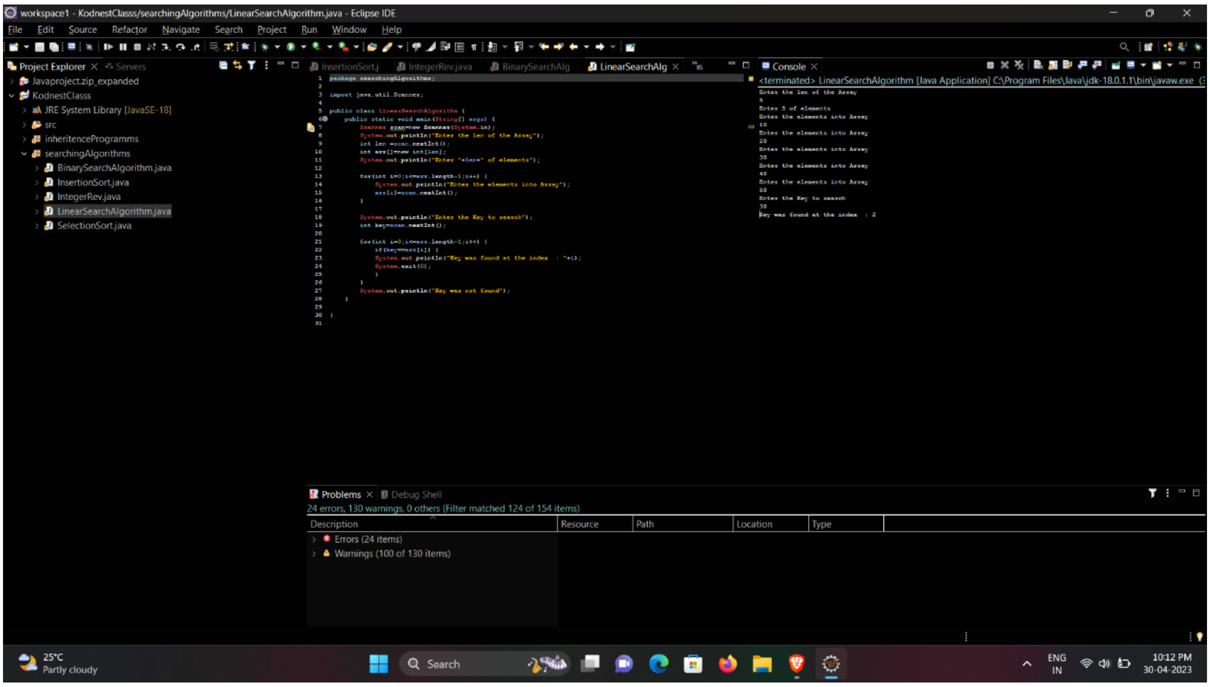
# Linear Search Program

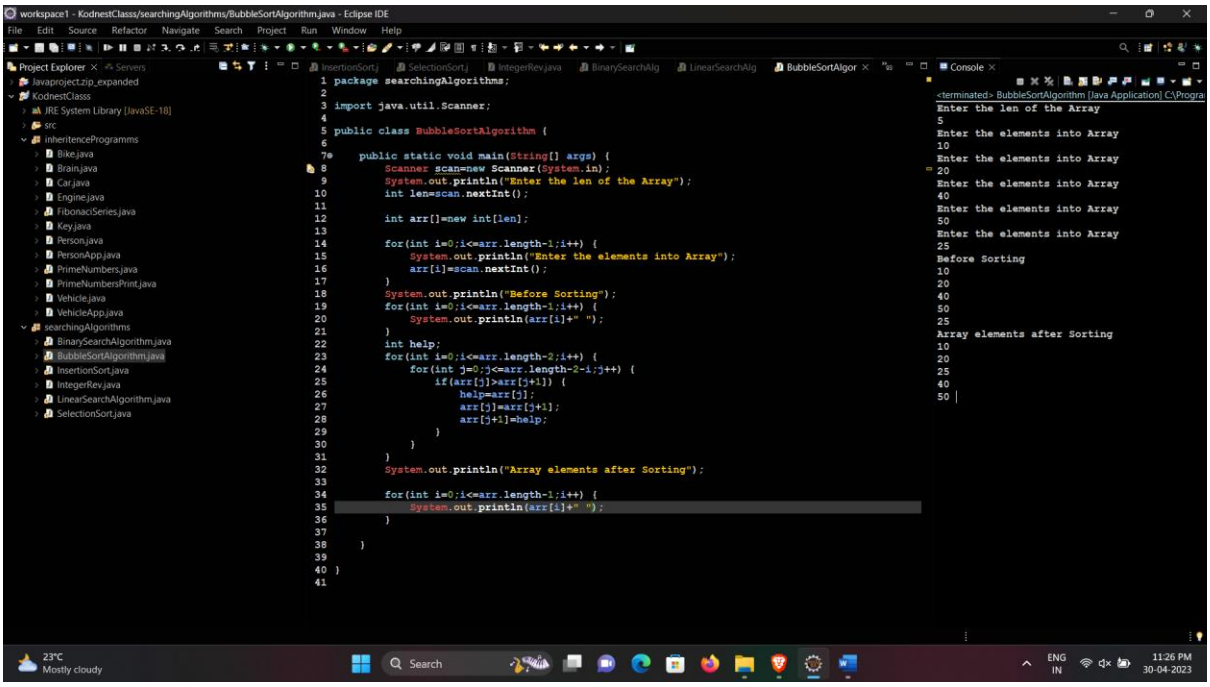


```java
package searchingAlgorithms;

import java.util.Scanner;

public class LinearSearchAlgorithm {
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter the len of the Array");
        int len =scan.nextInt();
        int arr[]=new int[len];
        System.out.println("Enter "+len+" of elements");

        for(int i=0;i<=arr.length-1;i++) {
            System.out.println("Enter the elements into Array");
            arr[i]=scan.nextInt();
        }

        System.out.println("Enter the Key to search");
        int key=scan.nextInt();

        for(int i=0;i<=arr.length-1;i++) {
            if(key==arr[i]) {
            System.out.println("Key was found at the index  : "+i);
            System.exit(0);
            }
        }
        System.out.println("Key was not found");
    }

}
```

```
<terminated> LinearSearchAlgorithm [Java Application] C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (
Enter the len of the Array
5
Enter 5 of elements
Enter the elements into Array
10
Enter the elements into Array
20
Enter the elements into Array
30
Enter the elements into Array
40
Enter the elements into Array
50
Enter the Key to search
30
Key was found at the index  : 2
```

```
24 errors, 130 warnings, 0 others (Filter matched 124 of 154 items)
> Errors (24 items)
> Warnings (100 of 130 items)
```

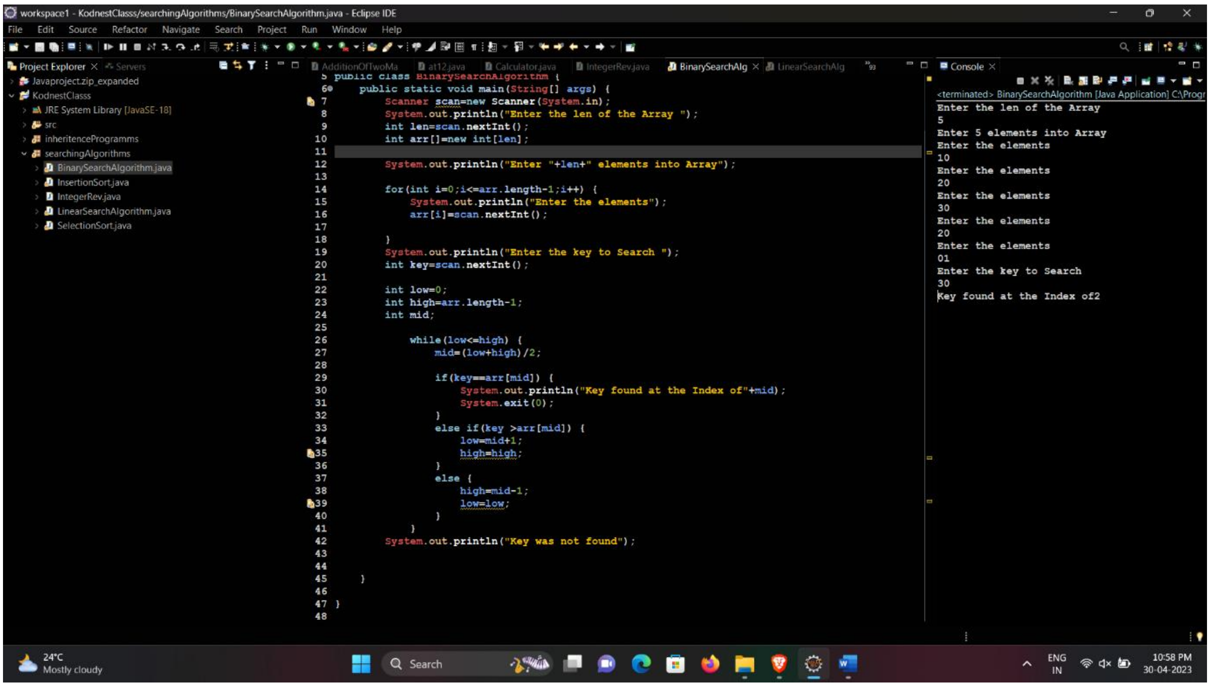
# Bubble Sort Program



```java
package searchingAlgorithms;

import java.util.Scanner;

public class BubbleSortAlgorithm {

    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter the len of the Array");
        int len=scan.nextInt();

        int arr[]=new int[len];

        for(int i=0;i<=arr.length-1;i++) {
            System.out.println("Enter the elements into Array");
            arr[i]=scan.nextInt();
        }
        System.out.println("Before Sorting");
        for(int i=0;i<=arr.length-1;i++) {
            System.out.println(arr[i]+" ");
        }
        int help;
        for(int i=0;i<=arr.length-2;i++) {
            for(int j=0;j<=arr.length-2-i;j++) {
                if(arr[j]>arr[j+1]) {
                    help=arr[j];
                    arr[j]=arr[j+1];
                    arr[j+1]=help;
                }
            }
        }
        System.out.println("Array elements after Sorting");

        for(int i=0;i<=arr.length-1;i++) {
            System.out.println(arr[i]+" ");
        }

    }

}
```

```
<terminated> BubbleSortAlgorithm [Java Application] C:\Progra
Enter the len of the Array
5
Enter the elements into Array
10
Enter the elements into Array
20
Enter the elements into Array
40
Enter the elements into Array
50
Enter the elements into Array
25
Before Sorting
10
20
40
50
25
Array elements after Sorting
10
20
25
40
50
```

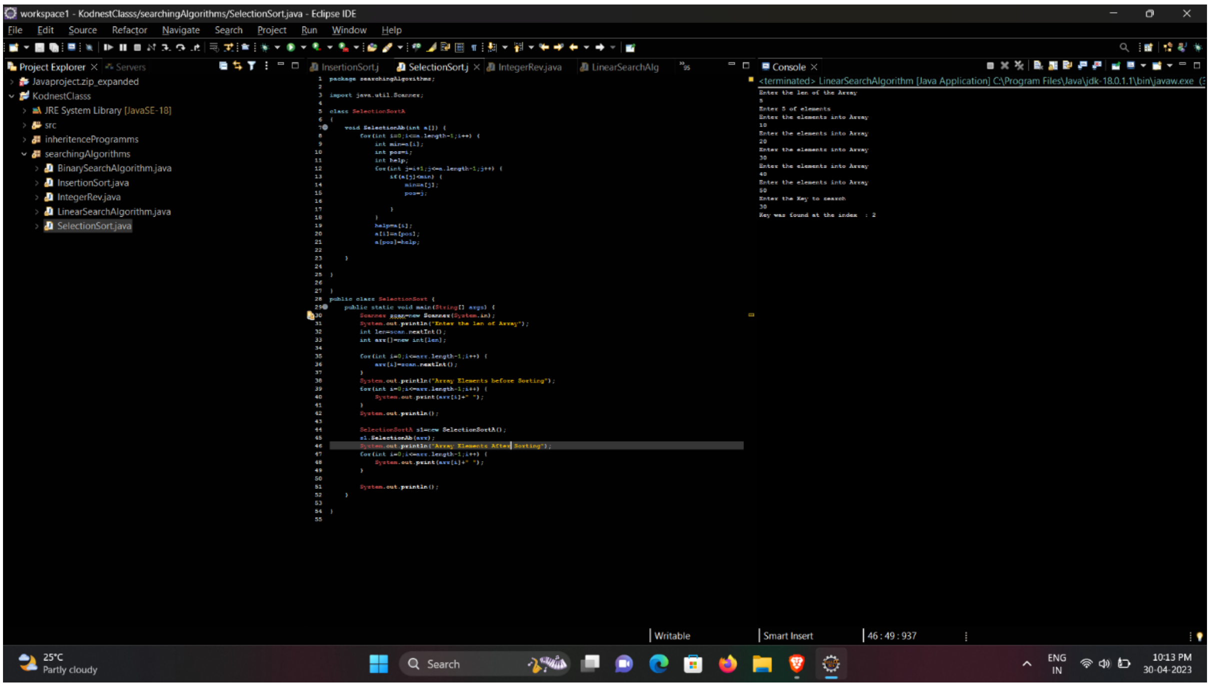
# Binary Search Program



```java
public class BinarySearchAlgorithm {
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter the len of the Array ");
        int len=scan.nextInt();
        int arr[]=new int[len];

        System.out.println("Enter "+len+" elements into Array");

        for(int i=0;i<=arr.length-1;i++) {
            System.out.println("Enter the elements");
            arr[i]=scan.nextInt();

        }
        System.out.println("Enter the key to Search ");
        int key=scan.nextInt();

        int low=0;
        int high=arr.length-1;
        int mid;

            while(low<=high) {
                mid=(low+high)/2;

                if(key==arr[mid]) {
                    System.out.println("Key found at the Index of"+mid);
                    System.exit(0);
                }
                else if(key >arr[mid]) {
                    low=mid+1;
                    high=high;
                }
                else {
                    high=mid-1;
                    low=low;
                }
            }
        System.out.println("Key was not found");


    }

}
```

```
<terminated> BinarySearchAlgorithm [Java Application] C:\Progr
Enter the len of the Array
5
Enter 5 elements into Array
Enter the elements
10
Enter the elements
20
Enter the elements
30
Enter the elements
20
Enter the elements
01
Enter the key to Search
30
Key found at the Index of2
```

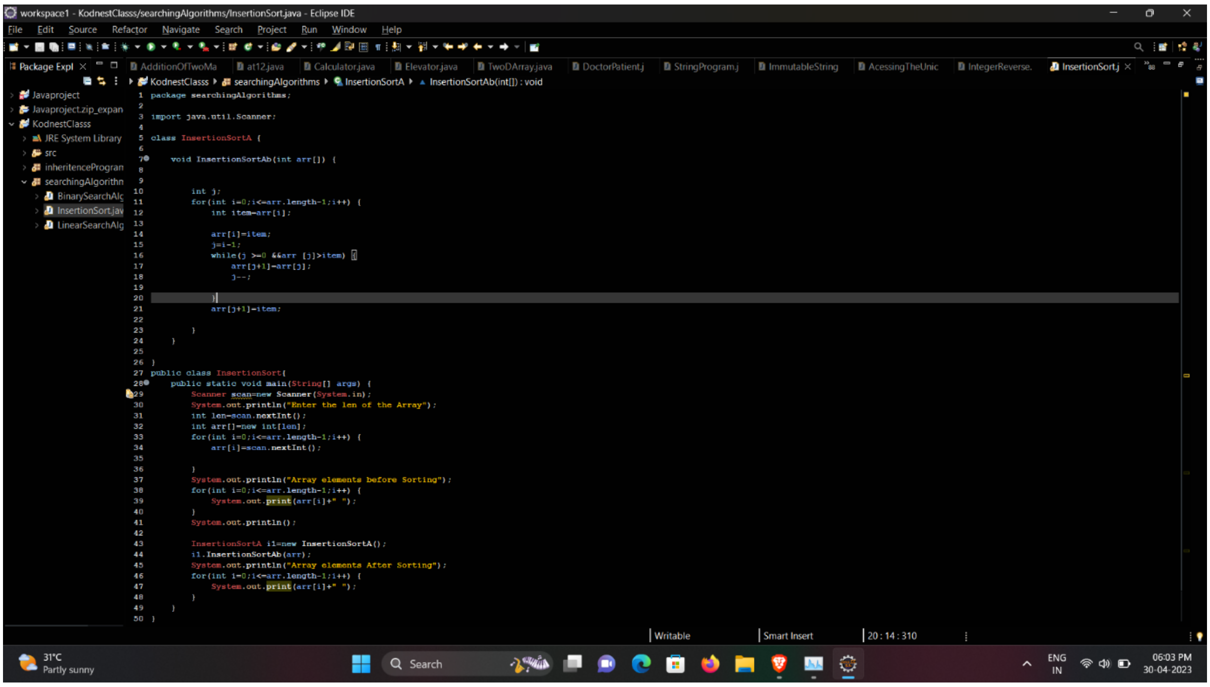
# Selection Sort Program



```java
package searchingAlgorithms;

import java.util.Scanner;

class SelectionSortA
{
    void SelectionAb(int a[]) {
        for(int i=0;i<=a.length-1;i++) {
            int min=a[i];
            int pos=i;
            int help;
            for(int j=i+1;j<=a.length-1;j++) {
                if(a[j]<min) {
                    min=a[j];
                    pos=j;

                }
            }
            help=a[i];
            a[i]=a[pos];
            a[pos]=help;

    }

}

}
public class SelectionSort {
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter the len of Array");
        int len=scan.nextInt();
        int arr[]=new int[len];

        for(int i=0;i<=arr.length-1;i++) {
            arr[i]=scan.nextInt();
        }
        System.out.println("Array Elements before Sorting");
        for(int i=0;i<=arr.length-1;i++) {
            System.out.print(arr[i]+" ");
        }
        System.out.println();

        SelectionSortA s1=new SelectionSortA();
        s1.SelectionAb(arr);
        System.out.println("Array Elements After Sorting");
        for(int i=0;i<=arr.length-1;i++) {
            System.out.print(arr[i]+" ");
        }

        System.out.println();
    }

}
```

```
<terminated> LinearSearchAlgorithm [Java Application] C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe  (3
Enter the len of the Array
5
Enter 5 of elements
Enter the elements into Array
10
Enter the elements into Array
20
Enter the elements into Array
30
Enter the elements into Array
40
Enter the elements into Array
50
Enter the Key to search
30
Key was found at the index  : 2
```

# Insertion Sort Program

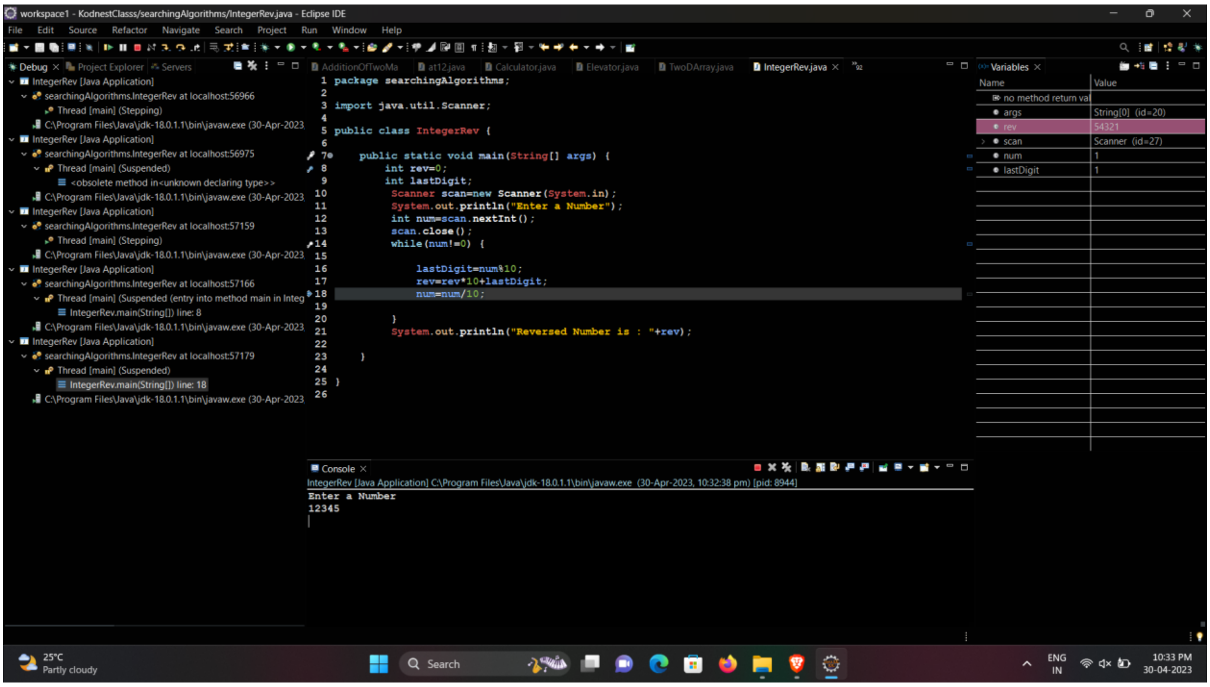


workspace1 - KodnestClasss/searchingAlgorithms/InsertionSort.java - Eclipse IDE

KodnestClasss ▸ searchingAlgorithms ▸ InsertionSortA ▸ InsertionSortAb(int[]) : void

```java
package searchingAlgorithms;

import java.util.Scanner;

class InsertionSortA {

    void InsertionSortAb(int arr[]) {


        int j;
        for(int i=0;i<=arr.length-1;i++) {
            int item=arr[i];

            arr[i]=item;
            j=i-1;
            while(j >=0 &&arr [j]>item) {
                arr[j+1]=arr[j];
                j--;

            }
            arr[j+1]=item;

        }
    }

}
public class InsertionSort{
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter the len of the Array");
        int len=scan.nextInt();
        int arr[]=new int[len];
        for(int i=0;i<=arr.length-1;i++) {
            arr[i]=scan.nextInt();

        }
        System.out.println("Array elements before Sorting");
        for(int i=0;i<=arr.length-1;i++) {
            System.out.print(arr[i]+" ");
        }
        System.out.println();

        InsertionSortA i1=new InsertionSortA();
        i1.InsertionSortAb(arr);
        System.out.println("Array elements After Sorting");
        for(int i=0;i<=arr.length-1;i++) {
            System.out.print(arr[i]+" ");
        }
    }
}
```

# Reverse Integer Debug mode



workspace1 - KodnestClasss/searchingAlgorithms/IntegerRev.java - Eclipse IDE

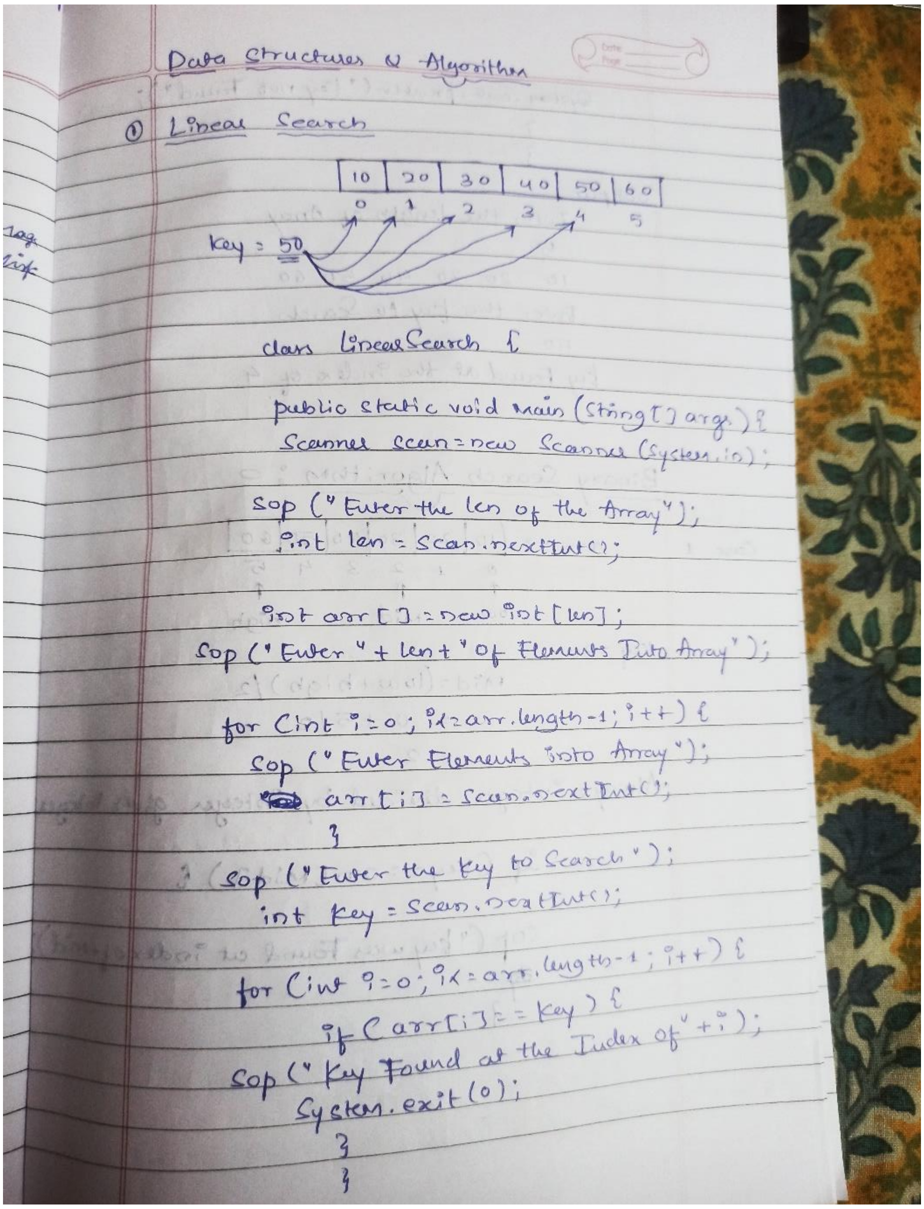
```
IntegerRev [Java Application]
  searchingAlgorithms.IntegerRev at localhost:56966
    Thread [main] (Stepping)
  C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (30-Apr-2023
IntegerRev [Java Application]
  searchingAlgorithms.IntegerRev at localhost:56975
    Thread [main] (Suspended)
      <obsolete method in<unknown declaring type>>
  C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (30-Apr-2023
IntegerRev [Java Application]
  searchingAlgorithms.IntegerRev at localhost:57159
    Thread [main] (Stepping)
  C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (30-Apr-2023
IntegerRev [Java Application]
  searchingAlgorithms.IntegerRev at localhost:57166
    Thread [main] (Suspended (entry into method main in Integ
      IntegerRev.main(String[]) line: 8
  C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (30-Apr-2023
IntegerRev [Java Application]
  searchingAlgorithms.IntegerRev at localhost:57179
    Thread [main] (Suspended)
      IntegerRev.main(String[]) line: 18
  C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (30-Apr-2023
```

```java
package searchingAlgorithms;

import java.util.Scanner;

public class IntegerRev {

    public static void main(String[] args) {
        int rev=0;
        int lastDigit;
         Scanner scan=new Scanner(System.in);
         System.out.println("Enter a Number");
         int num=scan.nextInt();
         scan.close();
         while(num!=0) {

             lastDigit=num%10;
             rev=rev*10+lastDigit;
             num=num/10;

         }
         System.out.println("Reversed Number is : "+rev);

    }

}
```

| Name | Value |
|---|---|
| no method return val | |
| args | String[0] (id=20) |
| rev | 54321 |
| scan | Scanner (id=27) |
| num | 1 |
| lastDigit | 1 |

IntegerRev [Java Application] C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (30-Apr-2023, 10:32:38 pm) [pid: 8944]

```
Enter a Number
12345
```



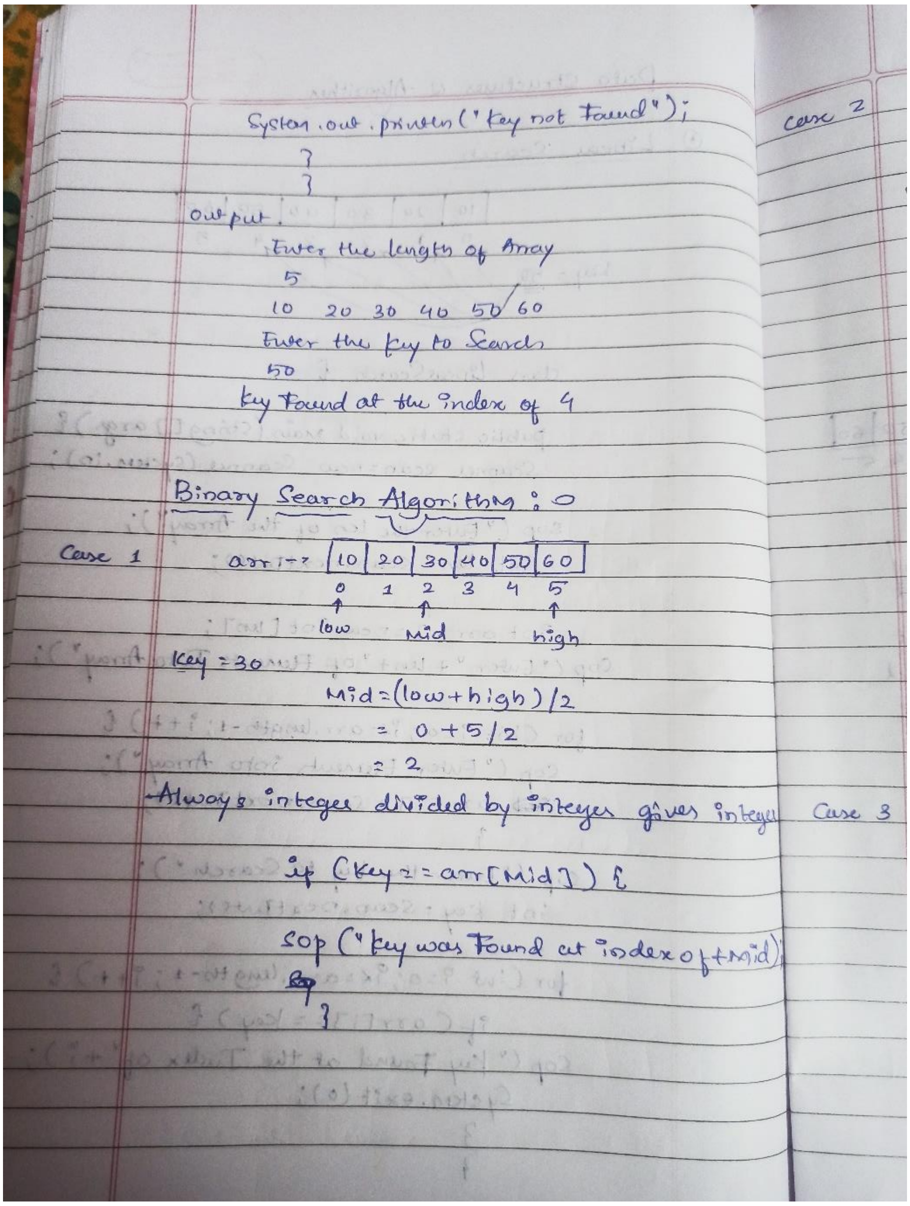
Data Structures & Algorithm

① Linear Search

| 10 | 20 | 30 | 40 | 50 | 60 |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 |

Key = 50

```
clars LinearSearch {

    public static void main (String[] args) {
    Scanner scan = new Scanner (System.in);

   sop ("Enter the len of the Array");
    int len = Scan.nextInt();

   int arr[] = new int [len];
 sop ("Enter " + len + " of Elements Into Array");

 for (int i=0; i<=arr.length-1; i++) {
   sop ("Enter Elements into Array");
      arr[i] = Scan.nextInt();
        }
  sop ("Enter the key to Search");
    int key = Scan.nextInt();

  for (int i=0; i<=arr.length-1; i++) {
        if (arr[i] == key) {
  sop (" Key Found at the Index of" + i);
      System.exit(0);
        }
        }
```



```
System.out.printen("key not Found");
        }
        }
```

output:

```
Enter the length of Array
5
10  20  30  40  50/60
Enter the key to Search
50
key Found at the index of 4
```

Binary Search Algorithm : 0

Case 2

Case 1

arr :->

| | | | | | |
|---|---|---|---|---|---|
| 10 | 20 | 30 | 40 | 50 | 60 |
| 0 | 1 | 2 | 3 | 4 | 5 |
| ↑ | | ↑ | | | ↑ |
| low | | Mid | | | high |

Key = 30

Mid = (low + high) / 2

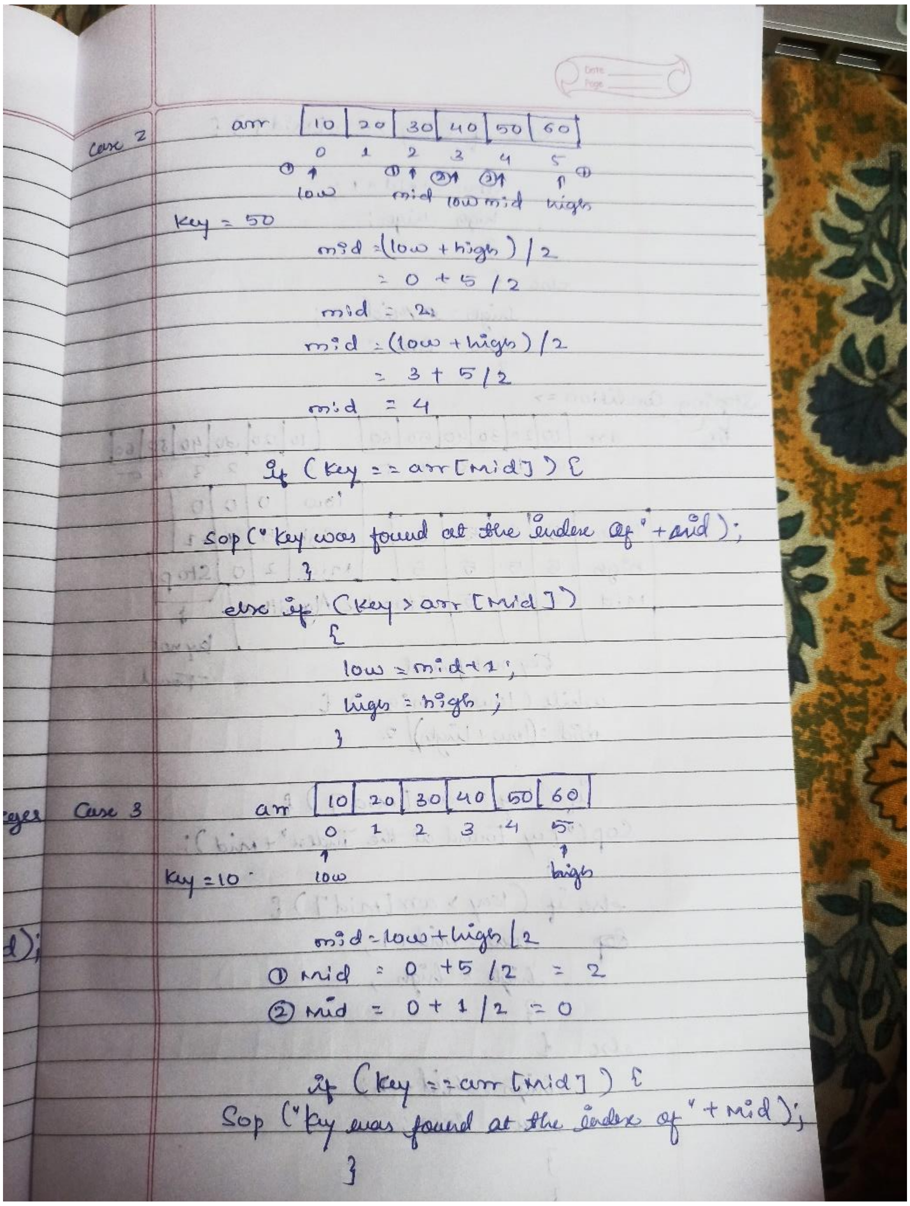
= 0 + 5/2

= 2

Always integer divided by integer gives integer

Case 3

```
if (Key == arr[Mid]) {
sop ("key was Found at index of +Mid);
~~Sop~~
}
```



Case 2

| arr | 10 | 20 | 30 | 40 | 50 | 60 |
|---|---|---|---|---|---|---|
|  | 0 | 1 | 2 | 3 | 4 | 5 |
|  | ① ↑<br>low |  | ① ↑<br>mid | ② ↑<br>low | ② ↑<br>mid | ↑ ①<br>high |

Key = 50

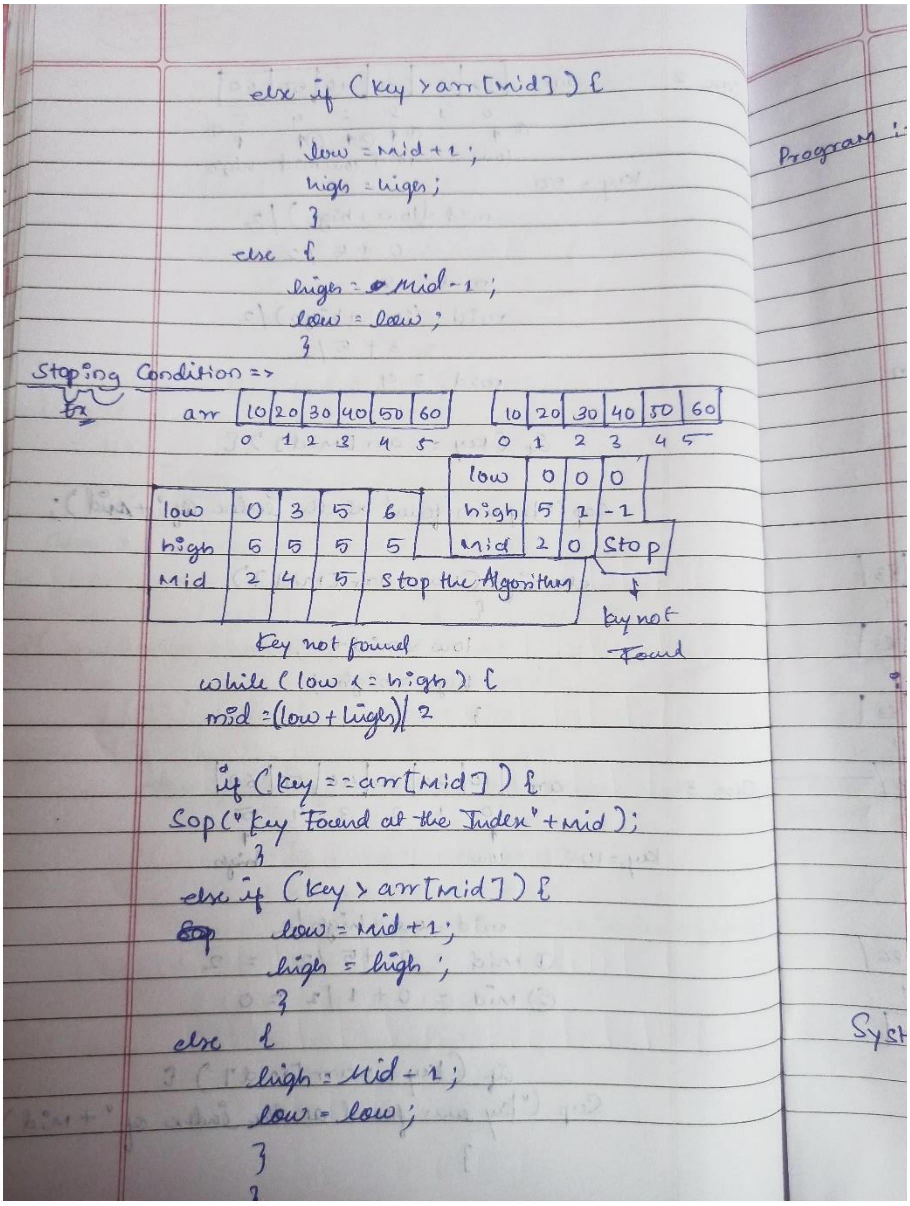
```
mid =(low + high ) / 2
    = 0 + 5 / 2
mid = 2
mid = (low + high) / 2
    = 3 + 5 / 2
mid = 4

if (Key == arr[mid]) {
sop (" Key was found at the index of " + mid);
}
else if (Key > arr [mid])
{
    low = mid+1;
    high = high ;
}
```

Case 3

| arr | 10 | 20 | 30 | 40 | 50 | 60 |
|---|---|---|---|---|---|---|
|  | 0 | 1 | 2 | 3 | 4 | 5 |
|  | ↑<br>low |  |  |  |  | ↑<br>high |

Key = 10

```
mid = low + high / 2
① mid = 0 + 5 / 2 = 2
② mid = 0 + 1 / 2 = 0

if (Key == arr [mid] ) {
Sop ("Key was found at the index of " + mid);
}
```



```
else if (key > arr[mid]) {
      low = mid + 1;
      high = high;
   }
else {
      high = mid - 1;
      low = low;
      }
```

Stoping Condition =>

Ex

arr

| 10 | 20 | 30 | 40 | 50 | 60 |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 |

| | | | | |
|---|---|---|---|---|
| low | 0 | 3 | 5 | 6 |
| high | 5 | 5 | 5 | 5 |
| Mid | 2 | 4 | 5 | Stop the Algorithm |

Key not found

| 10 | 20 | 30 | 40 | 50 | 60 |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 |

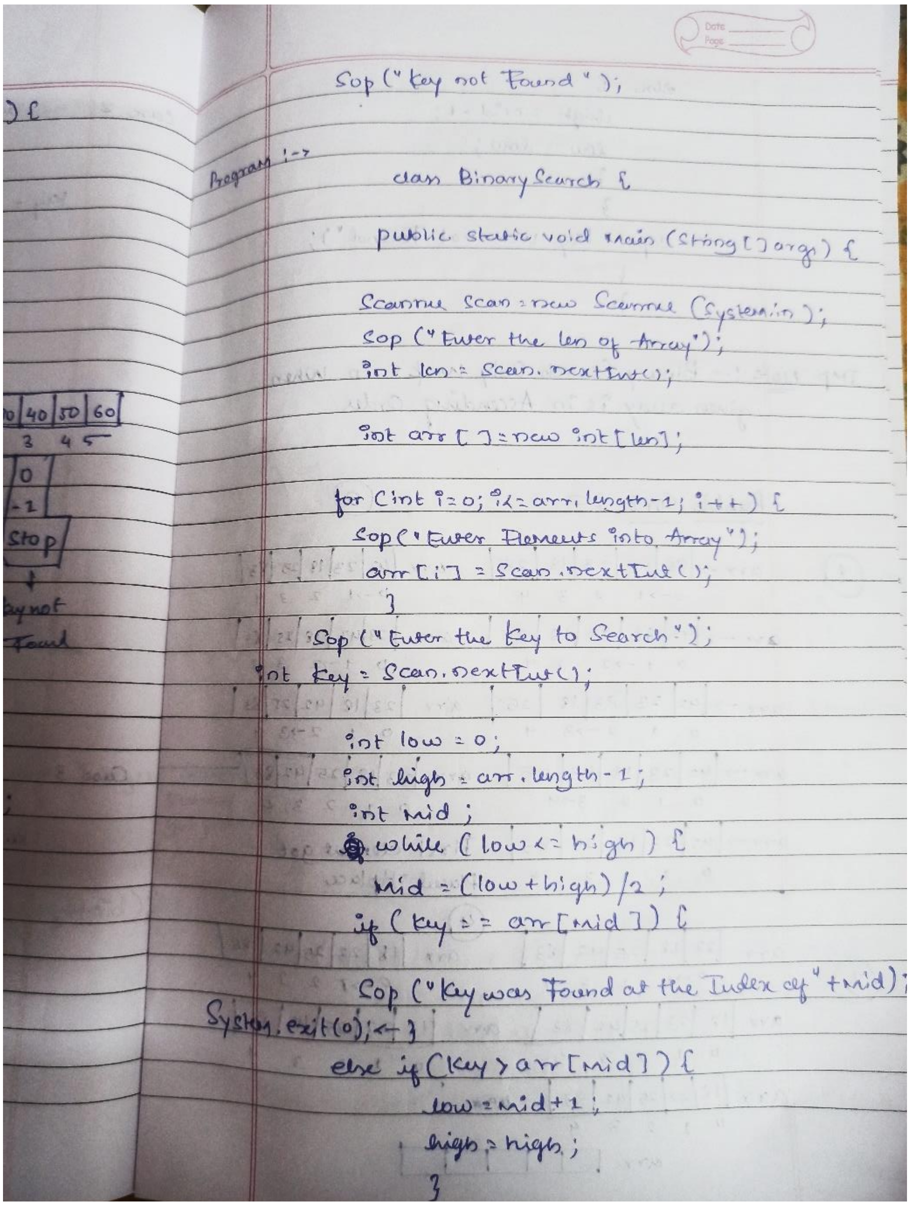
| | | | |
|---|---|---|---|
| low | 0 | 0 | 0 |
| high | 5 | 1 | -1 |
| mid | 2 | 0 | Stop |

key not Found

```
while (low <= high) {
mid = (low + high) / 2

  if (key == arr[mid]) {
Sop("key Found at the Index" + mid);
  }
else if (key > arr[mid]) {
      low = mid + 1;
      high = high;
   }
else {
      high = mid + 1;
      low = low;
   }
  }
```

Program :-

Syst



) {

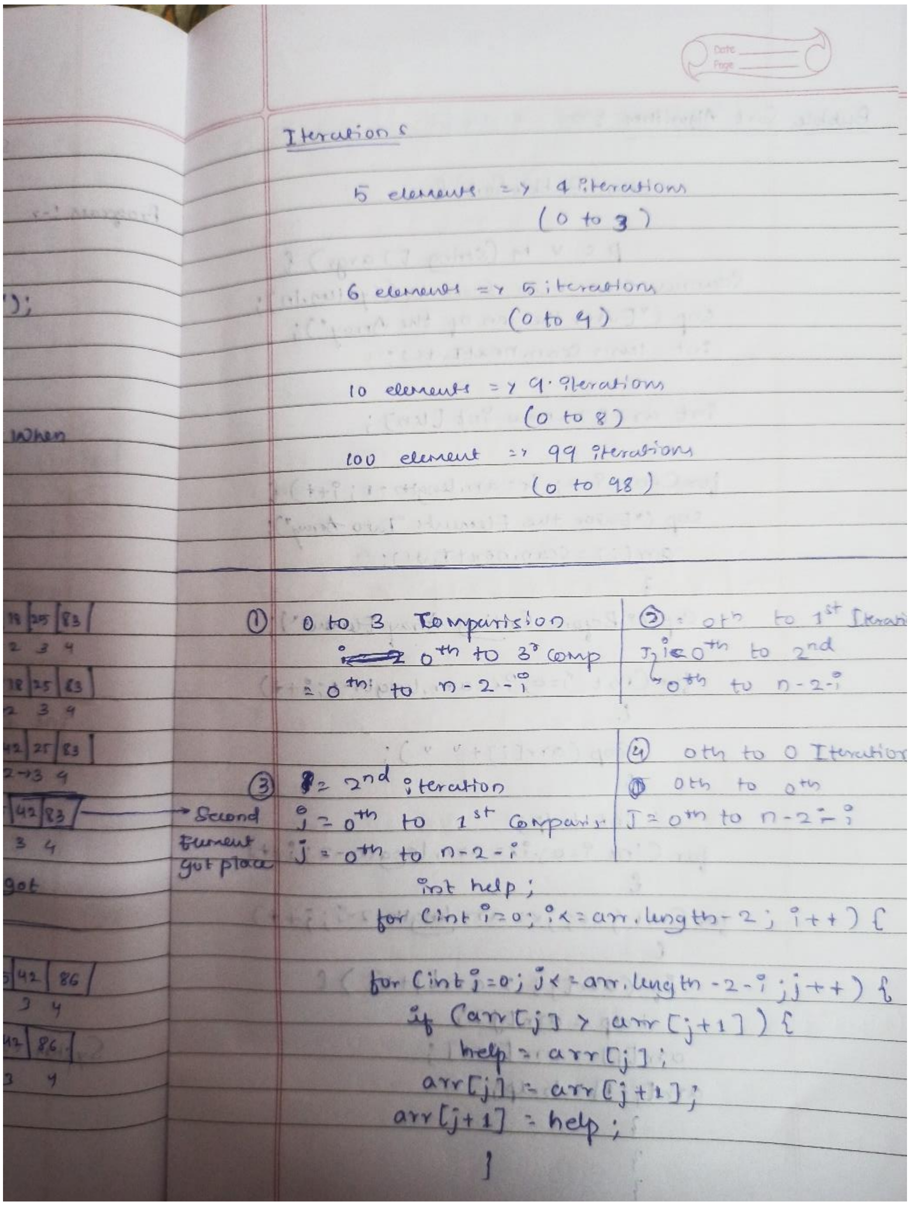
0 | 40 | 50 | 60
3 4 5
0
-1
stop
Key not Found

```
Sop ("Key not Found");

Program :->
        class BinarySearch {

         public static void main (String[] args) {

         Scanner scan = new Scanner (System.in);
         Sop ("Enter the len of Array");
         int len = Scan.nextInt();

         int arr [ ] = new int [len];

       for (int i=0; i<=arr.length-1; i++) {
         Sop ("Enter Elements into Array");
          arr[i] = Scan.nextInt();
           }
      Sop ("Enter the Key to Search");
  int Key = Scan.nextInt();

         int low = 0;
         int high = arr.length-1;
         int mid;
         while (low <= high) {
           mid = (low+high)/2;
         if (key == arr[mid]) {

           Sop ("Key was Found at the Index of" +mid);
System.exit(0); <- }
       else if (key > arr[mid]) {
              low = mid+1;
              high = high;
              }
        else {
            high = mid - 1;
            low = low;
            }
            }
        sop (" Key was not Found");
          }
          }
```

IMP Note :-> Binary Search Only works in When given array is in Ascending Order

**Bubble Sort :-**

```
①                                                  ②
arr -> | 86 | 42 | 23 | 18 | 25 |      arr | 42 | 23 | 18 | 25 | 83 |
         0->1   2    3    4                  0->1   2    3    4

arr -> | 42 | 83 | 23 | 18 | 25 |      arr | 23 | 42 | 18 | 25 | 83 |
         0    1->2    3    4                 0    1->2    3    4

arr -> | 42 | 23 | 83 | 18 | 25 |      arr | 23 | 18 | 42 | 25 | 83 |
         0    1    2->3    4                 0    1    2->3    4

arr -> | 42 | 23 | 18 | 83 | 25 |      arr | 23 | 18 | 25 | 42 | 83 | ——> Second
         0    1    2    3->4                 0    1    2    3    4      Element
                                                                        got place
arr -> | 42 | 23 | 18 | 25 | 83 | -> First element got
         0    1    2    3    4      Placed itsplace

                                                  ④
   arr | 23 | 18 | 25 | 42 | 83 |      arr | 18 | 23 | 25 | 42 | 86 |
③        0->1   2    3    4                  0->1   2    3    4

   arr | 18 | 23 | 25 | 42 | 83 |      arr | 18 | 23 | 25 | 42 | 86 |
         0    1->2    3    4                 0    1    2    3    4

   arr | 18 | 23 | 25 | 42 | 83 |
         0    1    2    3    4

         arr |    |    |    |    |    |
```



Iterations

5 element => 4 iterations
(0 to 3)

6 elements => 5 iterations
(0 to 4)

10 elements => 9 iterations
(0 to 8)

100 element => 99 iterations
(0 to 98)

① 0 to 3 Comparision
i ~~= 2~~ 0th to 3rd comp
= 0th to n-2-i

② : 0th to 1st Iterati
j, i = 0th to 2nd
→ 0th to n-2-i

③ i = 2nd iteration
j = 0th to 1st Comparis
j = 0th to n-2-i

④ 0th to 0 Iteration
① 0th to 0th
J = 0th to n-2-i

| 18 | 25 | 83 |
|---|---|---|
| 2 | 3 | 4 |

| 18 | 25 | 83 |
|---|---|---|
| 2 | 3 | 4 |

| 42 | 25 | 83 |
|---|---|---|
| 2→3 | 4 | |

| 42 | 83 |
|---|---|
| 3 | 4 |

→ Second Element got place

');

When

got

| 42 | 86 |
|---|---|
| 3 | 4 |

| 42 | 86 |
|---|---|
| 3 | 4 |

```
int help;
for (int i=0; i<=arr.length-2; i++) {

  for (int j=0; j<=arr.length-2-i; j++) {
      if (arr[j] > arr[j+1]) {
          help = arr[j];
        arr[j] = arr[j+1];
      arr[j+1] = help;
              }
```

Bubble Sort Algorithm :-

```
              class BubbleSort {
                  p s v m (String [] args) {
         Scanner scan=new Scanner (System.in);
           sop ("Enter the len of the Array");
           int  len = scan.nextInt();

           int arr[] =new int [len];

         for (int i=0; i<=arr.length-1; i++) {
            sop ("Enter the Elements Into Array");
              arr[i] = scan.nextInt();
              }
            sop ("Before Sorting Array Element");

            for (int i=0; i<=arr.length-1; i++)
                {
                Sop (arr[i]+" ");
                }
                int help;
         for (int i=0; i<=arr.length-2; i++)
              {
       for (int j=0; j<=arr.length-2-i; j++)
            {
             if (arr[j] > arr[j+1] ) {
              help = arr[j];
            arr[j] = arr[j+1];
          arr[j+1] = help;
                }
                }
                }
```

```
sop ("Array elements After Sorting");
        for (int i=0; i<=arr.length-1; i++)
            {
              sop (arr[i]+" ");
            }
            }
            }
```

Output

```
        Enter the len of Array
        5
        Enter elements Into Array
        86  42  23  18  25
        Array elements Before Sorting
        86  42  23  18  25
        Array elements After Sorting
        18  23  25  42  86
```

# Selection Sort :-

J is full down from Array

arr -> | 86 | 42 | 23 | 18 | 25 | J  min = 86 42 23 18

i = 0 j = 1 j = 2 j = 3 4   Pos = 0 1 2 3

(j++) (j++)

- 86 th position is 0 and i is handling that position

- Appoin a Servent called "j" One step ahead of 'i' to Search for real minum

j (j++) (j++)

i = 0 j = 1 j = 2 3 4 J

Element got placed it's position in the Array <- | 18 | 42 | 23 | 86 | 25 | min :- 42 23

pos :- ~~0~~ 1 2

J

arr -> | 18 | 23 | 42 | 86 | 25 | min ~~42~~ 25

0 1 i = 2 3 4   Pos ~~1~~ 4

J

arr -> | 18 | 23 | 25 | 86 | 42 | min = ~~86~~ 42

0 1 2 i = 3 4   pos = ~~3~~ 4

J

arr -> | 18 | 23 | 25 | 42 | 86 | min = last element got

0 1 2 3 i = 4   Pos = it's place Automatically

ay

86 42 23 18
0 1 2 3

lling

p ahead

2 23
2 2

25
4

ut got

ally

Psuedocode or Algorithm :-

```
for (int i=0; i<=n-2; i++) {

    min  = arr[i];
    Pos  = i;

  for (int j = i+1; j<=n-1; j++) {
     if (a[j] < min)
        {
        min = arr[j];
        pos = j;
        }
        }
  Swap a[i] with a[pos];
     }
```

Program 1 :->

```
        class Sorting {

     void selectionSort (int a[])
        {
         int min, pos, help;

  for (int i=0; i<=a.length-2; i++) {
       min  = a[i];
       pos  = i;

  for (int j=i+1; j<=a.length-1; j++) {
       if (a[j] < min) {
          min = a[j];
          pos = j;
```

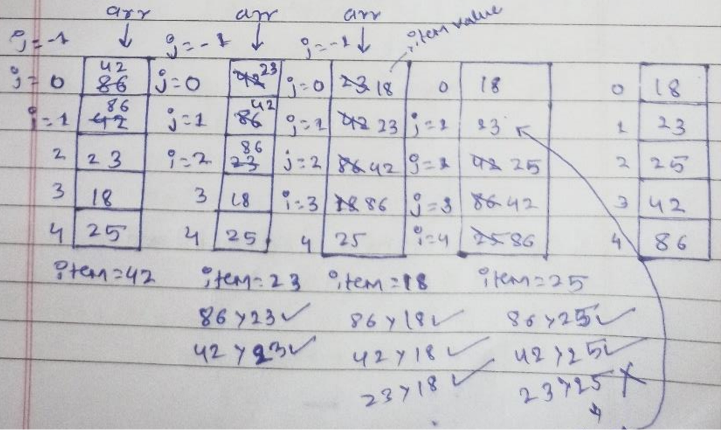
```
      } }
    help = a[i] ;
   a[i] = a[pos] ;
  a[pos] = help;
  }
  }
  }

class SortingApp {

     Psvm (String [] args) {

   Scanner scan = new Scanner (System.in);
   Sop ("Enter the len of the Array");
   int len = scan.nextInt();

   int arr[] = new int [len]

  for (int i=0; i<=arr.length-1; i++) {
       arr [i] = scan.nextInt();
     }
 Sop ("Array Content before Sorting ");
 for (int i=0; i<=arr.length-1; i++) {

   Sop (arr[i]+" ");
    }
   Sopln ();

Sorting S1 = new Sorting();
  S1.selectionSort (arr);
```

```
Sop ("Array Contents After Sorting");

 for (int i=0; i<=arr.length-1; i++) {

~~Sop~~   ~~(arr[i]) Scan.nextI~~
        Sop (arr[i]+" ");
          }
          }
          }
```

## Insertion Sort :-

Q. Write a Java program to sort the Content of an Integer array in assending Order by using insertion sort algorithm.?

Note:- The statements to sort an array using insertio sort Algorithm shoulb be written in a method by name insertionSort() Which would accept an inte array as parameters and would return void



```
logic :-> for (int i=1 ; i<= ~~arr.leng~~ n-1 ; i++) {

        item  = arr[i]
          j   = i-1;

    as long as ( j is +ve && arr[j] > item)
          {
          push a[j] one level below;
        a[j+1] = a[j];

          j--;
         }
        a[j+1] = item
        }
```

Date ______
Page ______

## Debug module :-

Q  Write a Java program to reverse the given number?

Ex  1 2 3 4 5        Output => 5 4 3 2 1

item)

```
10 ) 1 2 3 4 ( 1 2 3      10 ) 1 2 3 ( 1 2     10 ) 1 2 ( 1
     1 0 ↓                     1 0 ↓                1 0
     -----                     -----                ---
       2 3                       2 3                  2
       2 0 ↓                     2 0
       -----                     ---
         3 4                       3
         3 0                                    10 ) 1 ( 0
         ---                                         0
           4                                         ---
                                                     1
```

```
n = 1234
int digit;
rev = rev * 10 + lastDigit;
num = num/10;
  }
Sop(rev);
```

Program :→

```
        class Sample {

            P s v m (String [] args) {
          Scanner scan = new Scanner(System.in);
          sop ('Enter a Number');
          int num = Scan.nextInt();
            int lastgit, rev = 0;
              While (num != 0) {
      rev ~~lastDigit~~ = rev * 10 + lastDigit;
              num = num / 10;
             }
          Sop ('Reversed num' + rev);
```

```
Output
      Enter a Number
      1234
      Reversed num 4321
```

Another Way Using Method

```
        class Reverse {

            static int ReverseNum (int num)
              {
              int lastDigit;
              int rev =0;

            while (num !=0) {

             lastDigit = num % 10;
           rev = rev* 10 + lastDigit;
             num=num/10;
             }
         ~~Sop (Reverse~~
             return rev;
           } }
class RevApp {
          public static void main (String [] args)
            {
    Scanner scan= new Scanner(System.in);
     sop("Enter a Number");
     int num = scan.nextInt();
int rev = ReverseNumber.ReverseNum (num);
  sop ("Reversed Num"+ rev);
     }
     }
```

output

```
Enter A Number
1234
Reversed num is 4321
```

F6 -> will not take you inside the Method use F5 instead of F6

F5 -> will go inside the method.

Research { F7 ->
F8 ->

Debug Mode :-

It is a Mode in Eclipse IDE Which is used by the developers for debugging the program Code In this Mode the execution will happen line by line

Use of debugging

1) If Compilation is Sucessful but the program is not giving expeted Output then, to trace the Mistake in the program while execution debugging is used

2) To understand the Controlflow inside the program

Eclipse Debug Mode keys

① F6 => key is used to execute the Currently Selected line and to send Control to the Nextline in the program

## Documentation in Java :-

- As every Software would have documentation even Java (JDK) as a Software also has a Documentation This documentation can be used to know the things present in Java (JDK) and how to use Software Java (JDK)