**JAVA PROGRAMS**

Link:- https://www.javatpoint.com/java-programs#java-basic-programs

**Linear Search**

It sequentially checks each element of the list until a match is found or the whole list has been searched

package array.ty; //package

public class LinearSearch {

public static void main(String[] args) {

int[] arr= {23,14,5,46,30,8};

int search=30;

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

if(search==arr[i]) {

System.*out*.println("search ele present at index: "+i);

}

}

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Binary Search**

* Binary search works only for sorted array
* It will be splitted into half as position that count contain the element
* Low ,high , mid
* To calculate mid => mid=(low+high)/2;
* Binary search works until the low<=high

package array.ty;

public class BinarySearch {

public static int binary(int[] arr,int n,int searchEle) {

int low=0;

int high=n-1;

while(low<=high) {

int mid=(low+high)/2;

if(searchEle==arr[mid]) {

return mid;

}

else if(searchEle<arr[mid]) {

high=mid-1;

}

else if(searchEle>arr[mid]) {

low=mid+1;

}

}

return -1;

}

public static void main(String[] args) {

int[] arr= {05,14,18,21,26,28,30,58,70,85};

int n=arr.length;

int searchEle=30;

int mid=*binary*(arr,n,searchEle);

System.*out*.println("found at index: "+mid +" the element:"+ arr[mid]);

}

}

**Java Array Program**

1. Program to copy all elements of one array into another array

**package** array.ty;

**public** **class** CopyArrElementToAnother {

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

**int**[] arr= {1,2,3,4,5,6,7,8,9,10};

**int** n=arr.length;

**int** arr2[]=**new** **int**[n];

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

arr2[i]=arr[i];

}

**for**(**int** i=0;i<arr2.length;i++) {

System.***out***.print(arr2[i]+" ");

}

}

}

**----------------------------------------------------------------------------------**

**2.Frequency count of elements in array**

**package** array.ty;

**public** **class** FrequencyCount {

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

// **TODO** Auto-generated method stub

**int**[] arr= {1,1,1,1,1,2,2,2,2,23,23,100,200};

**int** fr[]=**new** **int**[arr.length];

**int** visited=-1;

**int** count=0;

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

count=1;

**for**(**int** j=i+1;j<arr.length;j++) {

**if**(arr[i]==arr[j]) {

count++;

fr[j]=visited;

}

}

**if**(fr[i]!=-1) {

fr[i]=visited;

}

}

}

}

**----------------------------------------------------------------------------------**

**3.Left Rotate of the array**

**package** array.ty;

**public** **class** LeftRotate {

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

// **TODO** Auto-generated method stub

**int**[] arr= {1,2,3,4,5};

**int** n\_time\_rotate=2;

**int** temp;

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

temp=arr[0];

**int** j;

**for**(j=0;j<arr.length-1;j++) {

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

}

arr[j]=temp;

}

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

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

}

}

}

**----------------------------------------------------------------------------------4. Right rotate of an array**

**package** array.ty;

**public** **class** RightRotate {

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

**int**[] arr= {1,2,3,4,5};

//5,1,2,3,4

**int** temp;

**int** no\_of\_rotate=3;

**for**(**int** i=0;i<no\_of\_rotate;i++) {

temp=arr[arr.length-1];

**int** j;

**for**(j=arr.length-1;j>=1;j--) {

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

}

arr[0]=temp;

}

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

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

}

}

}

**5. Find Duplicate Element in the array**

**package** array.ty;

**public** **class** DuplicateEleInArr {

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

**int**[] arr= {1,1,2,3,4,4,5};

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

**if**(arr[i]==arr[i+1]) {

System.***out***.println(arr[i]);

}

}

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**6. Remove Duplicate Element from the array**

**package** array.ty;

**public** **class** RemoveDuplicateEleInArr {

**public** **static** **int** RemoveDupli(**int**[] arr) {

**int**[] temp=**new** **int**[arr.length];

**int** j=0;

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

**if**(arr[i]!=arr[i+1]) {

temp[j++]=arr[i];

}

}

temp[j++]=arr[arr.length-1];

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

arr[i]=temp[i];

}

**return** j;

/\*for(int i=0;i<j;i++) {

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

}

return j;\*/

}

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

**int**[] arr= {1,2,2,3,4,5,5,6,6,6};

**int** n=*RemoveDupli*(arr);

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

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

}

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* NUMBERS**

**1.Reverse a Number**

**package** numbers;

**public** **class** ReversetheNumber {

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

**int** num=1234;

**int** reverse=0;

**while**(num!=0) {

**int** last=num%10;

reverse=reverse\*10+last;

num=num/10;

}

System.***out***.println(""+reverse);

}

}

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*2.**