ASSIGNMENT NO.2

ALGORITHMS

SEARCHING:

1.BINARY SEARCH:

**package** finTech;

// its binary search

**public** **class** BinSort {

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

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

**int** a[]= {4,5,6,7,8,9};

**int** key=2;

**int** end=a.length;

**int** start=0;

**int** mid,flag=0;

**while**(start<=end) {

mid=(start+end)/2;

**if**(a[mid]==key) {

System.***out***.println("element found in"+mid);

flag=1;

**break**;

}

**if**(a[mid]>key) {

end=mid-1;

}

**if**(a[mid]<key) {

start=mid+1;

}

}

**if**(flag==0) {

System.***out***.println("element not found");

}

}

}

O/P:

element found in3

2.LINEAR SEARCH:

**package** finTech;

**public** **class** LinSearch {

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

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

**int** a[]= {4,5,6,7,8,9};

**int** key=5;

**int** flag=0;

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

**if**(a[i]==key) {

System.***out***.println("element found in: "+i);

flag=1;

}

}

**if**(flag==0) {

System.***out***.println("element not found");

}

}

}

O/P:

element found in: 1

SORTING:

BUBBLE SORT

**package** finTech;

**public** **class** bubbleSort {

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

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

**int** a[]= {-5,45,0,9,-15};

**int** temp;

System.***out***.println("Step no: 1");

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

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

**if**(a[j-1]>a[j]) {

temp=a[j-1];

a[j-1]=a[j];

a[j]=temp;

}

**for**(**int** k=0;k<a.length;k++) {

System.***out***.print(" "+a[k]);

}

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

}

System.***out***.println("\nStep no:"+(i+2));

}

**for**(**int** k=0;k<a.length;k++) {

System.***out***.print(" "+a[k]);

}

}

}

O/P:

Step no: 1

-5 45 0 9 -15

-5 0 45 9 -15

-5 0 9 45 -15

-5 0 9 -15 45

Step no:2

-5 0 9 -15 45

-5 0 9 -15 45

-5 0 -15 9 45

-5 0 -15 9 45

Step no:3

-5 0 -15 9 45

-5 -15 0 9 45

-5 -15 0 9 45

-5 -15 0 9 45

Step no:4

-15 -5 0 9 45

-15 -5 0 9 45

-15 -5 0 9 45

-15 -5 0 9 45

Step no:5

-15 -5 0 9 45

-15 -5 0 9 45

-15 -5 0 9 45

-15 -5 0 9 45

Step no:6

-15 -5 0 9 45

INSERTION SORT:

**package** finTech;

**public** **class** InserSort {

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

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

**int** a[]={3,7,10,4,6,2};

**int** temp,j=0;

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

temp=a[i];

j=i-1;

//belowcheck elem pos in ordered list

**while**(j>=0 && a[j]>=temp) {

a[j+1]=a[j];

j--;

}

a[j+1]=temp;

}

**for**(**int** k=0;k<a.length;k++) {

System.***out***.print(""+a[k]+" ");

}

}

}

O/P:

2 3 4 6 7 10

QUICK SORT:

**package** finTech;

//not completed

**public** **class** QuickSort {

**void** display(**int** arr[]) {

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

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

}

}

**int** partition(**int** a[],**int** low,**int** high) {

**int** pivot=a[high];

**int** l=(low-1);

**int** temp;

**for**(**int** j=low;j<=(high-1);j++) {

// System.out.println("inside partition "+low+" "+high+" "+pivot);

**if**(a[j]<pivot) {

l++;

temp=a[l];

a[l]=a[j];

a[j]=temp;

}

}

temp=a[l+1];

a[l+1]=a[high];

a[high]=temp;

**return** (l+1);

}

**int**[] quick(**int** a[],**int** low ,**int** high) {

**if**(low<high) {

//System.out.println("inside quick "+low+" "+high+" ");

**int** pi=partition(a,low,high);

quick(a,low,pi-1);

quick(a,pi+1,high);

}

**return** a;

}

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

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

**int** a[]= {7,1,3,5,2,6,4};

QuickSort Q=**new** QuickSort();

Q.display(a);

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

**int** b[]= Q.quick(a, 0, a.length-1);

Q.display(a);

}

}

O/P: 7 1 3 5 2 6 4

1 2 3 4 5 6 7

SELECTION SORT:

**package** finTech;

**public** **class** SelSort {

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

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

**int** a[]= {10,4,7,3};

**int** l=a.length,temp;

**int** min;

**int** index=0;

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

min=a[i];

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

**if**(a[j]<=a[i]) {

min=a[j];

index=j;

}

}

temp=a[i];

a[i]=a[index];

a[index]=temp;

}

**for**(**int** k=0;k<a.length;k++) {

System.***out***.print(" "+a[k]);

}

}

}

O/P: 3 4 7 10