**Arrays**

* Performing Insertion, Search & Deletion operations.

Program

class ArrayApp

{

public static void main(String[] args)

{

long[] arr; // reference to array

arr = new long[100]; // make array

int nElems = 0; // number of items

int j; // loop counter

long searchKey; // key of item to search for

//--------------------------------------------------------------

arr[0] = 77; // insert 10 items

arr[1] = 99;

arr[2] = 44;

arr[3] = 55;

arr[4] = 22;

arr[5] = 88;

arr[6] = 11;

arr[7] = 00;

arr[8] = 66;

arr[9] = 33;

nElems = 10; // now 10 items in array

//--------------------------------------------------------------

for(j=0; j<nElems; j++) // display items

System.out.print(arr[j] + “ “);

System.out.println(“”);

//--------------------------------------------------------------

searchKey = 66; // find item with key 66

for(j=0; j<nElems; j++) // for each element,

if(arr[j] == searchKey) // found item?

break; // yes, exit before end

if(j == nElems) // at the end?

System.out.println(“Can’t find “ + searchKey); // yes

else

System.out.println(“Found “ + searchKey); // no

//--------------------------------------------------------------

searchKey = 55; // delete item with key 55

for(j=0; j<nElems; j++) // look for it

if(arr[j] == searchKey)

break;

for(int k=j; k<nElems-1; k++) // move higher ones down

arr[k] = arr[k+1];

nElems--; // decrement size

//--------------------------------------------------------------

for(j=0; j<nElems; j++) // display items

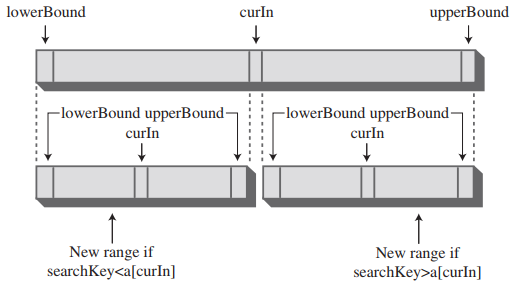
System.out.print( arr[j] + “ “);

System.out.println(“”);

} // end main()

} // end class ArrayApp

* Binary Search



public int find(long searchKey)

{

int lowerBound = 0;

int upperBound = nElems-1;

int curIn;

while(true)

{

curIn = (lowerBound + upperBound ) / 2;

if(a[curIn]==searchKey)

return curIn; // found it

else if(lowerBound > upperBound)

return nElems; // can’t find it

else // divide range

{

if(a[curIn] < searchKey)

lowerBound = curIn + 1; // it’s in upper half

else

upperBound = curIn - 1; // it’s in lower half

} // end else divide range

} // end while

} // end find()