Brandon Lee  
14010627X  
9/25/2014  
COMP2011  
Assignment 2

Report for Lab and Tutorial Two

1. The output of highArray.java:

11 32 57 66 108 39 19 0 66 33

Found 39

11 32 66 108 39 66 33

2. The output of selectSort.java:

7 9 11 23 44 6 19 0 55 33

0 6 7 9 11 19 23 33 44 55

3. The output of the new highArray.java program with the selection sort:

11 32 57 66 108 39 19 0 66 33

Found 39

11 32 33 39 66 66 108

Source code on next page:

Source Code:

/\* Brandon Lee

\* 9/25/14

\* COMP 2011

\* Assignment 2

\*/

// highArray.java

// demonstrates array class with high-level interface

////////////////////////////////////////////////////////////////

**class** HighArray

{

**private** **long**[] a; // ref to array a

**private** **int** nElems; // number of data items

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

**public** HighArray(**int** max) // constructor

{

a = **new** **long**[max]; // create the array

nElems = 0; // no items yet

}

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

**public** **void** selectionSort()

{

**int** out, in, min;

**for**(out=0; out<nElems-1; out++) // outer loop

{

min = out; // minimum

**for**(in=out+1; in<nElems; in++) // inner loop

**if**(a[in] < a[min] ) // if min greater,

min = in; // we have a new min

swap(out, min); // swap them

} // end for(out)

} // end selectionSort()

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

**private** **void** swap(**int** one, **int** two)

{

**long** temp = a[one];

a[one] = a[two];

a[two] = temp;

}

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

**public** **boolean** find(**long** searchKey)

{ // find specified value

**int** j;

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

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

**break**; // exit loop before end

**if**(j == nElems) // gone to end?

**return** **false**; // yes, can't find it

**else**

**return** **true**; // no, found it

} // end find()

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

**public** **void** insert(**long** value) // put element into array

{

a[nElems] = value; // insert it

nElems++; // increment size

}

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

**public** **boolean** delete(**long** value)

{

**int** j;

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

**if**( value == a[j] )

**break**;

**if**(j==nElems) // can't find it

**return** **false**;

**else** // found it

{

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

a[k] = a[k+1];

nElems--; // decrement size

**return** **true**;

}

} // end delete()

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

**public** **void** display() // displays array contents

{

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

System.***out***.print(a[j] + " "); // display it

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

}

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

} // end class HighArray

////////////////////////////////////////////////////////////////

**class** sample

{

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

{

**int** maxSize = 100; // array size

HighArray arr; // reference to array

arr = **new** HighArray(maxSize); // create the array

arr.insert(11); // insert 10 items

arr.insert(32);

arr.insert(57);

arr.insert(66);

arr.insert(108);

arr.insert(39);

arr.insert(19);

arr.insert(0);

arr.insert(66);

arr.insert(33);

System.***out***.print("Intitial array: ");

arr.display(); // display items

**int** searchKey = 39; // search for item

**if**( arr.find(searchKey) )

System.***out***.println("Found " + searchKey);

**else**

System.***out***.println("Can't find " + searchKey);

arr.delete(0); // delete 3 items

arr.delete(57);

arr.delete(19);

System.***out***.print("Unsorted array: ");

arr.display();

arr.selectionSort();

System.***out***.print("Sorted array: ");

arr.display(); // display items again

} // end main()

} // end class HighArrayApp