CS209 Lab04A
fa2014
dutta/streller

PROBLEM STATEMENT:

The declaration, creation, and use of arrays.
Inserting/removing items into/from an array.
Passing an array as a parameter to a method.
Performing operations on arrays.

CODE :

Obtain file ArrayDemoConsole.txt
Create a new project for this assignment. The project and folder should be named Lab04
Compile and run the ArrayDemoConsole.java program. The input/output generated
by the existing program consists of windows such as the following:

Modify/enhance the program.

Modification One.

Write a method named arrayToString.

The method must take two parameters: (1) an array of elements of type int and (2) a number of type int representing the number of elements actually inserted into the array.
The method must return a string that lists the elements of the array as one line of text, with a blank space separating the successive elements of the array. This string must be comprised of just the elements actually inserted into the array.
The returned string is intended for display purposes, to be printed to the Command Shell

Modification Two.

Replace existing code within main for-loop with a call to arrayToString method.

Modify the existing program code so that the new arrayToString method is called within the main for-loop to display the list of elements in each popup dialog after the text “Array elements:”. In other words, there is existing code that creates a string of text to list the array elements entered thus far (in the previous passes through the loop body). You are to replace this existing code with a call to the new method instead. Pass the array and element count into the new method as parameters.
As a result of your modification, the four lines of code shown below will be replaced by one line of code. The new one line will call the arrayToString method and assign the return value to variable arrayAStr.

```
arrayAStr = "";
int j
for ( j = 0; j < countA; j++)
{ arrayAStr += A[j] + " "; }
```

Modification Three ..

Modify the code that displays the final array information so that it uses the arrayToString method.

Modify the existing program code that follows after the main for-loop.
The arrayToString method must be called in the code after the for-loop finishes (below the for-loop). This code displays the final list of array elements.

On the first line, the output display should continue to show the total number of elements in the array.
On the second line, the display should show the phrase ?The array elements are:?, followed by the list of the actual elements, all on one line. That is, the display of the final list of array elements in the Output pane of NetBeans should still look the same as in the window capture above, with the same explanatory label, but the code must call the method arrayToString.
On the third line, the output should continue to display the sum of the array elements.

Modification Four

Add code to display a JOptionPane dialog after the for-loop.

The dialog should display all of the output listed above as part of Mod 3.
This same output string should be displayed to both the NetBeans Output pane and the newly added JOptionPane dialog.

Modification Fine

Create and populate array B.

n the main method of the program, declare and create a second array named B.
When declaring the new array B, the size of the array should be specified using the same symbolic constant as used for the existing array A. Do not declare another constant. That is, use the same unmodifiable (final) variable as is currently used to declare the size of array A.
Add another for-loop to populate the new array B. The new for-loop should get each array element for array B from the user using a JOptionPane popup dialog, just as for array A.
Each popup dialog must list the elements entered thus far for array B, just as for array A.
The arrayToString method must be called, as for array A.
Add the new for-loop for array B below the code that deals with array A.

Modification Six

Display the number of elements, contents, and sum for array B.

After the for-loop that populates array B (described above), add code to display the following (just like for array A):

The number of elements in the array.
The list of the elements in the array. Be sure to invoke the method arrayToString.
The sum of the elements in the array. Be sure to invoke the method arraySum.
Add code to display the results in a JOptionPane dialog as well as in the Output pane of NetBeans, just like for array A.

Modification Seven

Add code for the selection sort method.

The method must take an integer array as a parameter. It has no return. See the code for the insertion sort method in your text in Chapter 6. It is found in the problem section at the end of the chapter. Place the code appropriately in this class.

Modification Eight

Before the final printout for the arrays, call the sort method for each array.

Modification Nine

Present a final display of all the program results (output) in a JOptionPane dialog in addition to the Output pane of NetBeans. In this final JOptionPane dialog, display all the information that was previously presented on both arrays including the number of elements, list of elements, and sum of elements for each of arrays A and B. Be sure to invoke methods arrayToString, arraySum and insertionSort for each array.

Format the source code properly:
Reformat the Java code so that it has proper indentation
and vertical and horizontal alignment.
Be sure to have a blank line right above each method definition.

Compile and debug your modified program.

Test your program using a sufficient variety of data to make sure that it works on all cases.
Test using a mix of positive, negative, and zero integers.

Run the program and make the following window captures.

Be sure to include a window capture of each of the different windows used in your program, including:
One of the JOptionPane dialogs that prompt for entry of an element for array A, after at least three elements have been entered.
Two of the JOptionPane dialogs that prompt for entry of an element for array B, after at least four elements have been entered.
Each of the JOptionPane dialogs that present output information on the arrays, including the final dialog that is presented.
The Output pane of NetBeans showing the complete final output of the program.
Examples of these windows are shown below.

As usual create ReadMe.pdf file.

Create a file named ReadMe.pdf
In this document, insert your name at the top, and on the next line insert the assignment number
Then enter any comments regarding the assignment and your program.
Then insert several window captures of windows showing the inputs and outputs from the execution of the program.
Be sure the ReadMe file is within your top level project folder.

Zip the project folder and all its contents.

Change the name the zip file so that its name consists of your name along with the assignment number, as follows: " LastName_ Lab_04A_cs209.zip".
Do not use spaces in the name of the file, use underscores or hyphens instead.

Deliverables:

Send to streller@ecc.edu an email this the exact subject

cs209__ Lab_04A

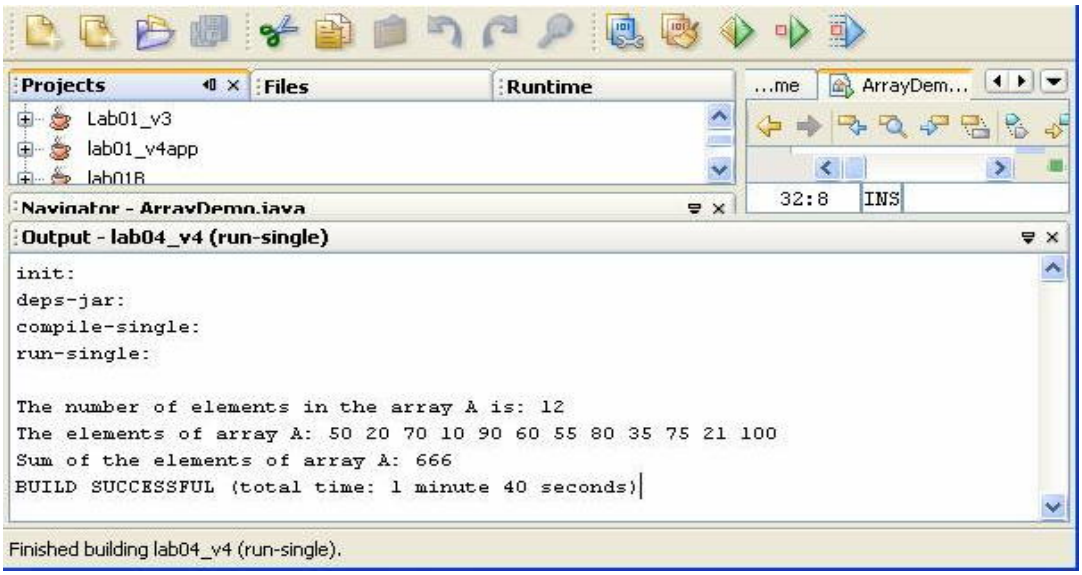
In this email attached the above named zip file

LastName_ Lab_04A_cs209.zip

Due Date : 25 September 2014 7:35pm

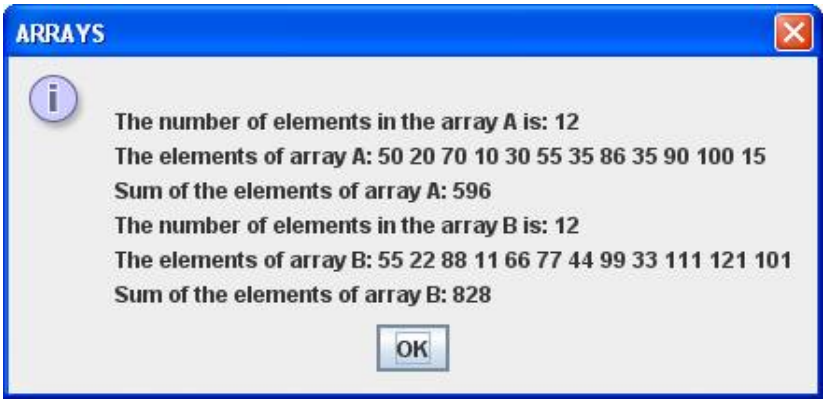

Input
Array elements:
Enter integer #0 to insert into first array A
50
OK
Cancel

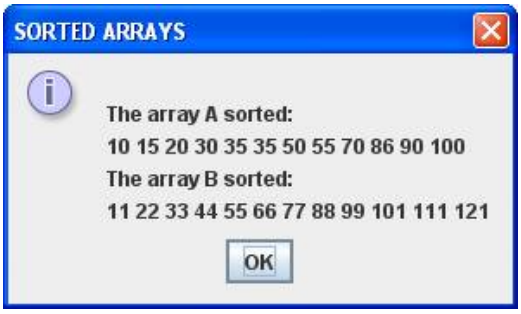

Input
Array elements: 50 20 70 10 90 60 55 80 35 75 21
Enter integer #11 to insert into first array A
100
OK
Cancel

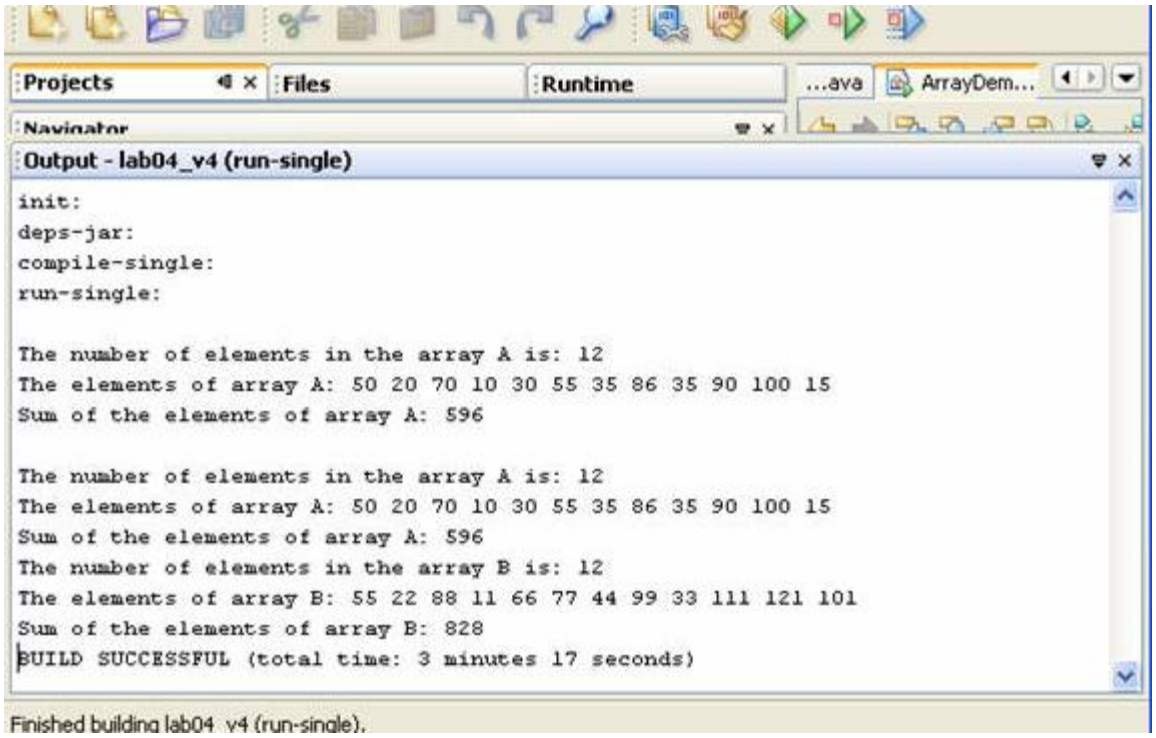

Projects
Files
Runtime
Lab01_v3
lab01_v4app
lab01B
...me
ArrayDem...
32:8
INS
Navigator - ArrayDemo.java
Output - lab04_v4 (run-single)
init:
deps-jar:
compile-single:
run-single:
The number of elements in the array A is: 12
The elements of array A: 50 20 70 10 90 60 55 80 35 75 21 100
Sum of the elements of array A: 666
BUILD SUCCESSFUL (total time: 1 minute 40 seconds)
Finished building lab04_v4 (run-single).


Input
Array elements:
Enter integer #0 to insert into first array A
50
OK
Cancel


Input
Array elements: 55 22 88 11 66 77 44 99 33 111 121
Enter integer #11 to insert into second array B
101
OK
Cancel


ARRAYS
The number of elements in the array A is: 12
The elements of array A: 50 20 70 10 30 55 35 86 35 90 100 15
Sum of the elements of array A: 596
The number of elements in the array B is: 12
The elements of array B: 55 22 88 11 66 77 44 99 33 111 121 101
Sum of the elements of array B: 828
OK


SORTED ARRAYS
The array A sorted:
10 15 20 30 35 35 50 55 70 86 90 100
The array B sorted:
11 22 33 44 55 66 77 88 99 101 111 121
OK


Projects
Files
Runtime
...ava
ArrayDem...
Navigator
Output - lab04_v4 (run-single)
init:
deps-jar:
compile-single:
run-single:
The number of elements in the array A is: 12
The elements of array A: 50 20 70 10 30 55 35 86 35 90 100 15
Sum of the elements of array A: 596
The number of elements in the array A is: 12
The elements of array A: 50 20 70 10 30 55 35 86 35 90 100 15
Sum of the elements of array A: 596
The number of elements in the array B is: 12
The elements of array B: 55 22 88 11 66 77 44 99 33 111 121 101
Sum of the elements of array B: 828
BUILD SUCCESSFUL (total time: 3 minutes 17 seconds)
Finished building lab04_v4 (run-single).


Input
Array elements: 50 20 70 10 30 55 35 86 35 90 100
Enter integer #11 to insert into first array A
15
OK
Cancel