

CANDIDATE NAME: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Are you looking for a new opportunity to have your work transform a business? Then look no further than Armedia

Please review the following questions and Provide All Requested Items of information for each question.

We understand that this may be a difficult evaluation test and take some time to complete. We ask you provide answers to all questions. If you are struggling, complete as much as possible for the question. It is imperative we understand your ability and we will review them with you at your in-person interview.

We at Armedia are looking for exceptional talent with folks that take pride in their work and strive to go above and beyond to reach the next level of excellence; as they will have the opportunity to work with some of the finest minds in this industry on cutting edge technology.

Thank you for considering Armedia! A great place to work to make a difference

1. Show the exact output produced by the following code segment.

char[][] pic = new char[6][6];

for (int i = 0; i < 6; i++)

for (int j = 0; j < 6; j++) {

if ( i == j || i == 0 || i == 5 )

pic[i][j] = '\*';

else

pic[i][j] = '.';

}

for (int i = 0; i < 6; i++) {

for (int j = 0; j < 6; j++)

System.out.print(pic[i][j]);

System.out.println();

}

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

.\*....

..\*...

...\*..

....\*.

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

2. What letters get written to the standard output with the following code?

class anyClass {

public static void main(String[] args) {

try {

clsMethod1();

} catch (Exception e) {}

}

static void clsMethod1() {

try {

clsMethod2();

System.out.println(“a”);

} catch (ArithmeticException e) { System.out.println(“b”); }

Finally {System.out.println (“c”); }

System.out.println(“d”);

}

Static void clsMethod2() {

Throw new NullPointerException();

}

3. What output would you get when you invoke the following main classes?

class anyClass1 {

public static void main(String[] args) {

String s1 = new String (“hello”);

String s2 = new String (“hello”);

If (s1.equals(s2)) System.out.println(“equal”);

Else System.out.println(“not equal”);

}

}

class anyClass2 {

public static void main(String[] args) {

StringBuffer sb1 = new StringBuffer (“hello”);

StringBuffer sb2 = new StringBuffer (“hello”);

If (sb1.equals(sb2)) System.out.println(“equal”);

Else System.out.println(“not equal”);

}

}

(please note java keywords letters need to be in lower case: finally, static…)

c

Programming Tests – Please completed the Following Exercises

4. EX1

Write a complete subroutine that finds the largest value in an array of ints. The subroutine should have one parameter, which is an array of type int[]. The largest number in the array should be returned as the value of the subroutine.

public class myClass3 {

public myClass3() {

}

public static int findLargest(int[] nums) {

int largest = nums[0];

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

if(nums[i] > largest) {

largest = nums[i];

}

}

return largest;

}

public static void main(String args[]) {

int[] ary = { 55, 89, 90, 23, 44, -1, -6, 9};

int largest = myClass3.findLargest(ary);

System.out.println(largest);

}

}

5. EX2

import java.util.\*;

public class BetterProgrammerTask {

public static int[] retainPositiveNumbers(int[] a) {

/\*

Please implement this method to

return a new array with only positive numbers from the given array.

The elements in the resulting array shall be sorted in the ascending order.

\*/

}

}

package armedia;

/\*

Please implement this method to

return a new array with only positive numbers from the given array.

The elements in the resulting array shall be sorted in the ascending order.

\*/

public class myClass4 {

// TODO Auto-generated constructor stub

public static int[] retainPositiveNumbers(int[] nums) {

int[] positive = new int[nums.length];

for(int i = 0, j = 0; i< nums.length; i++, j++) {

if(nums[i] > 0) {

positive[j] = nums[i];

}

}

return positive;

}

public static void main(String args[]) {

int[] ary = { 55, 89, 90, 23, 44, -1, -6, 9};

int[] positive = myClass4.retainPositiveNumbers(ary);

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

System.out.println(positive[i]);

}

}

}

6. EX3

What output is produced by the following program segment? Why?

String name;

int i;

boolean startWord;

name = "Richard M. Nixon";

startWord = true;

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

if (startWord)

System.out.println(name.charAt(i));

if (name.charAt(i) == ' ')

startWord = true;

else

startWord = false;

}

7. EX4

Write a program to determine whether the SUM of the Digits of the Number to the Power of the Length of the number equals the original number.

For example, given a number ‘XYZ’, which is 3 digit number - calculate if X3 + Y3 + Z3 = XYZ

For example, given a number ‘UXYZ’, which is 4 digit number - calculate if U4 + X4 + Y4 + Z4 = UXYZ

Find all Numbers between 10-999 that satisfy this condition (All one digit numbers will always satisfy this condition so let’s ignore them)A client would like some processing and calculations done based on the following requirements. Please write a program that processes the listed data and displays the results as indicated in the requirements. Please provide the output from the execution of the code.

8. EX5

Create a function with the following input parameters to perform the task defined

findMatch(int[] list, int total, int numToUse) {

/\* list = list of integers

total = a single integer

numToUse = how many numbers from the list to use as a maximum

This function should print all combinations of values in ‘list’, when summed up return the value in ‘total’

The value provided in ‘numToUse’ indicates the maximum values to use from ‘list’ to calculate the ‘total’. If the value is 2 then use a maximum of 2 elements from ‘list’. If the value is 3, use a maximum of 3 elements from ‘list’

For example:

list[1,5,6,3,6,7]

total = 13

numToUse = 2

Then the following should be printed:

Element 3 + Element 6 = 13

Element 5 + Element 6 = 13

If numToUse = 3 with the same inputs, then

Element 3 + Element 6 = 13

Element 5 + Element 6 = 13

Element 1 + Element 2 + Element 6 = 13

\*/

}

Execute for the following inputs:

list = {3,4,6,7,10,3,9,15,17,17, 10,7}

total = 20

numToUse = 2, 3, and 4

Please provide the output from the execution of the code.

static void findMatch(int[] list, int total, int numToUse) {

for(int i=0; i < list.length && numToUse > 0; i++) {

for(int j=1; j < list.length && numToUse > 0; j++)

{

if(list[i] + list[j] == total ) {

System.out.println(" Element " + i + " Element = " + j + " == " + total);

numToUse-- ;

};

}

}

System.out.println();

System.out.println();

System.out.println();

}

public static void main (String args[]) {

int[] list = {3,4,6,7,10,3,9,15,17,17, 10,7};

int total = 20;

int numToUse = 2;

findMatch(list, total, numToUse);

numToUse = 3;

findMatch(list, total, numToUse);

numToUse = 4;

findMatch(list, total, numToUse);

}

9. EX. 6

We have a process where we receive data in a file which are required to process manually. We would like to automate this through the use of a program that will automatically read the data and process it, and then display the results to us in a report. Please provide the output from the execution of the code.

The format of the input file and the sample data to use is shown below.

11

\*2 2

2 3

4 5

\*2 2

3 11

11 3

\*4 2

4 4

5 5

6 6

55 99

\*3 3

3 5 a

4 7 a

3 4 b

\*3 2

3 3

4 5 c a

3 5 b

\*3 3

4 2 d

3 4 a

3 3 b

\*3 3

4 2 a

3 3 a

4

\*4 2

2 2

3 3

4 4

\*1 3

-1 -2 a

\*4 2

0 0

1 -1

-1 1

1111111 222222

\*3 3

\*2 3

The data is structured as follows.

The first record in the file will indicate the number of data samples to read.

The subsequent records will provide the specific data to process

The first record must contain two items that identify the data group to follow – this can be considered the control record – the items are separated by spaces

The first character of the control record must be a ‘\*’

The first number indicates the number of data records in the data group

The second number indicates the number of elements in the data group

The next set of records will be the actual data that needs to be processed and will contain any number of elements separated by spaces. You only need to use the applicable values.

If there is a discrepancy in the data counts, the entire set of data is to be rejected and an error message printed indicated which data group the error is in.

If there is any error, generate an error message and reject the entire data group.

Blank lines should not be processed

You are to process the data as follows:

If the data contains two parameters, perform the following calculations

Calculate the average of the two values and display with the following label “AVG = “

Calculate the results of the following formula and display with the following label “FORM = “

Result = pi \* (first value) + psi \* (1.44 + (second value)3 )

Calculate the average of the ‘Result’ calculated above from all the data points

The results must be displayed to 3 significant digits – nnnn.000

Display the total number of records reads

If the data contains three parameters

Calculate the average of the three values and display with the following label “AVG 33 = “

Calculate the results of the following formula and display with the following label “FORM 33 = “

Result = pi \* (first value) + psi \* (1.44 + (second value)3 ) + (zeta (lookup using the following mapping)) / (second value)

a 🡺 3

b 🡺 4

c 🡺 6

If a translation is not found, then use ‘a’ as the default.

Calculate the average of the ‘Result’ calculated above from all the data points

The results must be displayed to 3 significant digits – nnnn.000

Display the total number of records reads

The constants pi,psi, and zeta are defined as follows:

Pi = 3.14

Psi = 6.39485

Zeta = 3.2

You are more than welcome to add additional data to validate other error and test conditions. All code and output should be sent back so it can be reviewed and verified. If there are any questions, please feel free to contact us.

AVG = 3.500

FORM = 220.918

AVG = 6.500

FORM = 853.893

AVG = 8.500

FORM = 8562.942

AVG = 12.500

FORM = 249.178

AVG = 6.000

FORM = 463.807

AVG = 7.500

FORM = 857.033

AVG = 9.000

FORM = 1442.104

AVG = 104.500

FORM = 6205131.237

AVG 33 = 5.500

FORM 33 = 824.538

AVG 33 = 7.500

FORM 33 = 2219.883

AVG 33 = 5.000

FORM 33 = 454.113

AVG 33 = 5.000

FORM 33 = 696.334

AVG 33 = 5.000

FORM 33 = 436.091

AVG = 4.500

FORM = 224.058

AVG 33 = 5.500

FORM 33 = 838.956

AVG 33 = 5.500

FORM 33 = 1032.733

AVG 33 = 5.500

FORM 33 = 824.538

AVG 33 = 5.000

FORM 33 = 89.311

AVG 33 = 5.000

FORM 33 = 436.091

AVG 33 = 4.500

FORM 33 = 226.242

AVG 33 = 4.500

FORM 33 = 549.203

AVG 33 = 4.500

FORM 33 = 202.212

AVG 33 = 5.000

FORM 33 = 89.311

AVG 33 = 4.500

FORM 33 = 202.212

AVG = 3.000

FORM = 99.415

AVG = 4.500

FORM = 224.058

AVG = 6.000

FORM = 463.807

AVG 33 = -2.000

FORM 33 = -61.474

AVG = 0.000

FORM = 41.977

AVG = 0.500

FORM = 38.722

AVG = -0.500

FORM = 45.231

AVG = 1222222.000

* FORM = 70176469857746104.000