1. Write code that will instantiate (create) a *double* array called *sgt* having 800 elements.

double[] sgt = new double[800];

2. double []dfw = new double[21];

System.out.println( dfw.length ); //What is printed?

21

3. For the code in #1 above, write a *for*-loop that will cycle through all the elements of

*double sgt[]* and store the square root of the index of each element in that element.

for (int i = 0; i < 800; i++) {

sgt[i] = Math.sqrt(sgt[i]);

}

4. Assuming *rtl* is the name of an array, what’s wrong with this code?

double rtl\_len = rtl.length( );

rtl.length returns an int not a double.

5. On **one** line of code, both declare a character array called *cr* and initialize its elements

to be ‘a’, ‘b’, ‘c’, ‘d’, and ‘e’.

ch[] cr = {‘a’, ‘b’, ‘c’, ‘d’, ‘e’}

6. Refer to #5 above. What is the value of *cr.length* ?

5

7. Write code that will print the sum of the squares of all the elements of the *ref* integer

array.

int sum = 0;

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

sum += (ref[i] \* ref[i]);

}

System.out.println(sum);

8. What’s wrong with the following code?

for (int k = 2; k < homer.length; k++)

{

homer[k +1] = k;

}

The for loop will iterate past homer’s bounds.

9. Fill in the blanks below to enable us to pass a *double* array called *dbx* to a method

called *heroWorship*. Within the method, the array should be called *vb*.

boolean bbc = heroWorship(dbx);

public boolean heroWorship(double vb)

10. Assume the five values an integer array *adc* contains are: 34, 56, -102, 18, and 5.

What is the value of *adc[1]* ?

56

11. Using the *adc* array from #10 above, what would be the value of *adc[3] + adc[4]*?

23

12. Using the *adc* array from #10 above, what would be the value of *adc[5]* ?

it doesn’t exist

13. Describe what the following code segment does:

for (int j = 0; j < b.length; j++)

b[j] = Math.abs( b[j] );

It turns all of the elements in the array positive.

14. For the *int* array *c = {1, 2, 3, 4}*, what would be the output of the following code?

String ss = “>>>”;

int len = ss.length( );

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

ss+= c[j];

System.out.println(ss);

>>>1234

15. Write a loop that locates the first occurrence of a negative integer in an array, *pg*.

When the loop is finished, the variable *indx* should contain the index of the negative

number, or the length of the array if there were no negative numbers in the array.

int indx = 0;

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

if (pg[i] < 0 ) {

break;

} else {

indx++;

}

}

16. String wc = “Whooping crane”;

String sp[] = wc.split(“oo”);

for(int j = 0; j < sp.length; j++)

{

System.out.println(sp[j]);

}

Wh

ping crane

17. List the elements of *String [ ]sArray = “fee fi fo ”.split(“*\\*s+”);*.

fee fi fo

18. List the elements of *String [ ]sp = “One two”.split(“Q”);*.

One two

19. Using the *split* method, write code that will count all of the occurrences of “th”

(without regard to upper or lower case) in “The best THERE is is Barth”.

String s1 = “The best THERE is is Barth”;

String[] sp = s1.split(th);

System.out.println(sp.length);