

1. (10 Points) What is the output of the following code:

```
public static void main(String[] args) {
    int s = 1;
    int t = 1;
    for ( int i = 1 ; i < 21 ; i += 2 ) {
        s = s + i;
        for ( int j = i ; j > 0 ; j -= 3 ) {
            t = t + (i - j);
        }
        s = s + t;
        System.out.println("t = " + t);
    }
    System.out.println("s = " + s);
}
```

2. (20 Points) Given the following classes, show the output from running the Test class and give an explanation:

<pre>public class Vehicle { private String name = new String("No Vehicle Name"); public Vehicle() { System.out.println("New Vehicle: " + name); } public Vehicle(String name) { this.name = new String(name); System.out.println("New Vehicle: " + name); } public void drive() { System.out.println("Vehicle drive: " + name); } }</pre>	<pre>public class SportsCar extends Car { private String name = new String("No Sports Car Name"); public SportsCar() { super("No Sports Car Name"); System.out.println("New SportsCar: " + name); } public SportsCar(String name) { super(); this.name = new String(name); System.out.println("New Vehicle: " + name); } public void drive() { System.out.println("Sports Car drive: " + name); } }</pre>
<pre>public class Car extends Vehicle { private String name = new String("No Car Name"); public Car() { super("No Car Name"); System.out.println("New Car: " + name); } public Car(String name) { super(); this.name = new String(name); System.out.println("New Car: " + name); } public void drive() { System.out.println("Car drive: " + name); } }</pre>	<pre>public class Question2 { public static void main(String args[]) { Vehicle v, v1, v2; Car c, c1, c2; SportsCar sc, sc1, sc2; v1 = new Vehicle("Veronica"); v2 = new Vehicle(); c1 = new Car("Carlos"); c2 = new Car(); sc1 = new SportsCar("Sport"); sc2 = new SportsCar(); v1.drive(); v2.drive(); c1.drive(); c2.drive(); sc1.drive(); sc2.drive(); v = c1; v.drive(); c = sc1; c.drive(); } }</pre>

3. (30 Points) Given an array of `String`. Each element in the array contains a `String` object. Example:

"Sameh"	"Eliot"	"Sarah"	"Dalia"	"Nidal"
---------	---------	---------	---------	---------

Write a method with the following signature:

```
public static String combine(String[] strings)
```

Which combines the `Strings` to form one `String` that would contain the last letters of each `String`, starting with the last name, followed by the second to last letter from each string again starting with the last name, etc.... You may assume that all `Strings` have the same length, but you may not assume that the array only has 5 entries. The above array would return the following `String`:

"lahthai aedlr imiaa laNDSSES"

4. (10 Points) Write a `for` loop to compute the sum $1^1 + 2^2 + 3^3 + 4^4 + 5^5 + \dots + n^n$. Assume that `n` is a variable that has already been defined.
5. (10 Points) Show the output from running the Test class:

```
public class Question5 {
    public static void main(String args[]) {
        String s1 = new String("josue");
        String s2 = new String("joseph");
        String s3 = new String("Jose");
        String s4 = s2;

        if (s1.substring(0,3).equals(s3.substring(0,3))) {
            System.out.println("Test1 Is A Success");
        } else {
            System.out.println("Test1 Is A Failure");
        }

        if (s2.substring(1,4).equals(s3.substring(1,4))) {
            System.out.println("Test2 Is A Success");
        } else {
            System.out.println("Test2 Is A Failure");
        }

        if (s2 == s4) {
            System.out.println("Test3 Is A Success");
        } else {
            System.out.println("Test3 Is A Failure");
        }

        if (s3.toLowerCase().substring(0,3).equals(s1.substring(0,3))) {
            System.out.println("Test4 Is A Success");
        } else {
            System.out.println("Test4 Is A Failure");
        }
    }
}
```

6. (40 Points) Write a complete Java class named `MyClass` that has the following `private` attributes:
- `myInts`, an array of `int`'s that has a maximum capacity of 100.
 - `numInts`, an `int` variable that keeps track of the number of elements in `myInts`.

And the following methods:

- `public MyClass()` – Constructor that initializes `myInts` and `numInts`.
- `public int addInt(int i)` – Adds `i` to `myInts` and updates `numInts`. Returns the index where `i` was added. If there is no room in the array, expand the array by adding another 100 spaces.
- `public int findFirst(int i)` – Find the first occurrence of `i` in `myInts` and return its index, return -1 if not found.
- `public int findLast(int i)` – Find the last occurrence of `i` in `myInts` and return its index, return -1 if not found.
- `public int getInt(int i)` – Return the integer at index `i` if it exists, return -9999 otherwise.
- `public boolean isFull()` – Returns true if `myInts` is full, false otherwise.
- `public boolean isEmpty()` – Returns true if `myInts` is empty, false otherwise.