Final Group Project

(To developed **in groups of 2 or 3 students**)

ACME Corporation and AJAX Inc. are important companies in the IT sector who have developed advanced sorting libraries to help other companies to facilitate the development of information systems. These solutions are implemented as Software as a Service (SaaS). As these methods are supposed to be executed in a SaaS architecture, each company has a different pricing model to access and execute their systems. ACME and AJAX have the following pricing model for their sorting products:

AJAX CORP (Cost per sort request)	**Cost**
up to 100,000 daily requests	$1,000 USD per request
Between 100,001 to 1,000,000 requests daily requests	$10,000 USD per request
(each request can sort up to 1000 records)	

ACME CORP (Cost per item to sort)	**Cost**
up to 100 items to sort per request	$1.00 USD per request
up to 1000 items to sort per request	$700 USD per request
up to 1,000,000 items to sort per request	$13,000 per request

The sorting platform available at CSS process the following data per day on average:

	MON	**TUES**	**WED**	**THURS**	**FRI**	**SAT**	**SUN**
# of daily records to sort (in millions)	10	1	30	50	0.5	100	0.1

Your current manager at Columbus Software Solutions (CSS) has asked your development team to check which of the two solutions (ACME or AJAX) is better for CSS to buy or use to implement the new sorting platform at CSS (maybe a combination of both?). Both companies have instrumented carefully modified versions of their algorithms for CSS (as .jar libraries) to study and decide what to buy.

Your manager has provided the following information about these libraries:

1. Each company has provided a .jar file with a carefully instrumented method that returns the number of instructions each sorting method executes. The methods are as follows:
   a. ACMESorting.jar. This jar file contains a package called acmesorting that contains only once class called ACMESorting. The ACMESorting class has only one static method with the following signature:

      public static int acmeSorting(int array[])

      Given the array passed as parameter, this method sorts the array and returns the number of instructions that the method executed to sort the array.

   b. AJAXSorting.jar: This jar file contains a package called ajaxsorting that contains only once class called AJAXSorting. The AJAXSorting class has only one static method with the following signature:

      public static int AJAXSort(int array[])

      Given the array passed as parameter, this method sorts the array and returns the number of instructions that the method executed to sort the array.

2. ACME Corporation claims that they have the best sorting algorithm in the market. They say the complexity of their algorithm is in the order of **O(log$_2$(n))**. AJAX Inc. claims that they have a better sorting algorithm than ACME Corp. but they have not provided the complexity of their sorting algorithm since it may reveal trade secrets that they do not want ACME Corp. or any other corporation to know about.

How CSS should implement their new sorting platform?

Deliverables:

1. A technical report explaining your analysis. Look for examples of technical reports in the Internet. The report must contain information about the analysis of the algorithms, the solution to implement to sort for CSS Corporation and an analysis of your proposed solution to CSS (in running time and monetary cost).

   Begin your tech report explaining and including a description on software as a service (SaaS), examples/scenarios in which companies (both small and big) have successfully used this technology.and drawbacks on using this type of technology. (15 points, one report per group)

2. Any source code you use to conduct experiments to find out more about each of the SaaS solutions. You must experiment! There is no way to develop your report without experimenting with the algorithms. (30 points, one submission per group)

3. Your coding solution to perform the sorting per day (there are 7 days on which CSS perform sorting per day). For each of the days create a class called CSSSorting, and for each day create a method, something like:
   a. public static void CSSSortMonday(int[] values),
   b. public static void CSSSortTuesday(int[] values),
   c. public static void CSSSortWednesday(int[] values),
   d. public static void CSSSortThursday(int[] values)
   e. public static void CSSSortFriday(int[] values),
   f. public static void CSSSortSaturday(int[] values),
   g. public static void CSSSortSunday(int[] values)

   In each of these methods implement how you think the sorting should take place based on the companies' libraries. (30 points, one submission per group. Must implement sorting for each day and a driver that shows a test on each day)

4. A presentation recorded in a video and uploaded to YouTube about your project. You can create a power point presentation and record your presentation explaining your report to your manager. You can use the freely available software OBSProject available at https://obsproject.com/ to record your presentation and upload it to YouTube. Also upload the presentation to the submission folder. (15 points, one video per group)

5. A submission of the **We Solve It! Report**. Please see the rubric, as well as hints on how to answer the report in the assignment's dropbox. (Submitted individually, 10 points).

Your code solutions/experimental code must be submitted as a Netbeans projects (for Dr. Perez class).

Total: 100 points max.

1. **Technical report: 15 points**
2. **Source code of experiments: 30 points**
3. **Coding solution: 30 points**
4. **Video: 15 points**
5. **QEP report: 10 points**