

Question 4 Not yet answered	John is writing a new class. That class has 5 methods but one of those methods (method A) has some fundamental logic that John doesn't want to be overriden by any subclass. What should John do?
Marked out of 1.00 Flag question	Select one: a. define the class as final b. defined method A as final
	c. there is nothing that he can do about d. defined method A as static
Question 5 Not yet answered Marked out of 1.00 Flag question	A subclass of which of the following classes should be used when trying to read data from a text file? Select one: File InputStream OutputStream Reader
Question 6 Not yet answered Marked out of 1.00 Flag question	Which of the following is the root class (apex) of the exception heirarchy in Java? Select one: Throwable Exception Error RuntimeException
Question 7 Not yet answered Marked out of 1.00 Flag question	Which of the following statements best describes the difference between <code>java.util.Date</code> and <code>java.sql.Date?</code> Select one: There is no difference; they both hold the same information. java.sql.Date represents the date without time information. java.sql.Date does not exist. java.sql.Date holds information in nanoseconds. java.sql.Date holds the database's time zone information.
Question 8 Not yet answered Marked out of 1.00 Flag question	Which of the following prefixes can be added to a number in order to indicate a binary literal? Select one or more: ✓ 0b ✓ 0B ○ oA ○ 0S ○ 0s

Question 9

Not yet answered

Marked out of 1.00



Which of the following phrases best describes what would happen if you attempted to compile and run code that calls the method in the snippet below on Java 7?

```
public String getSeasonUsingSwitchStatement(String seasonOfYearArg) {
    String nameOfSeason;
    switch (seasonOfYearArg) {
    case "May";
    case "May";
    nameOfSeason = "Spring";
    hard nameOfSeason = "Spring";
    case "July";
    case "July";
    case "August";
    nameOfSeason = "Summer";
    hard nameOfSeason = "Summer";
    break;
    case "Cutober";
    case "November";
    case "November";
    nameOfSeason = "Autumn";
    break;
    case "Season = "Autumn";
    case "Season = "Winter";
    case "February";
    case "February";
    case "February";
    case "February";
    case "Tebruary";
    c
                                                                                                                                                                                                :
throw new IllegalArgumentException("Invalid month for season of the year:" + seasonOfYearArg);
                                                                                                     }
return nameOfSeason;
```

Select one:

- $\bigcirc \ \ compilation\ error, because the switch statement does not call the getSeasonUsingSwitchStatement() method$
- ompilation error, because a String object is used in the expression of the switch statement
- ompilation error, because IllegalArgumentException is used instead of java.lang.Exception
- ompilation error because the variable nameOfSeason may not have been initialized



none of the above, the code will compile correctly

Question 10

Not yet answered

Marked out of 1.00



Flag question

We've deployed new release on the production. And it became very slow. What are your suggestion how to analyze and find bottleneck?

Select one or more:



M Check the logs



- Turn on monitoring tools
- Enable debug logging
- Add more instances to the cluster
- Call SaasOps



Check the difference in source code from previous version

Pray your favorite God.



Question 11

Not yet answered

Marked out of 1.00



From the perspective of a Java generational garbage collector, all new objects are considered to be part of:

Select one:

old generation



- PermGen
- Metaspace

Question 12

Not yet answered

Marked out of 1.00



If you need to find one item in a large unsorted collection but only once. What is faster way?

Select one:

Sort the collection using quick sort and get the element by binary search

Iterate through all entries and compare each item

Question 13

Not yet answered

Marked out of 1.00

Flag question

We can improve the performance of a system in which we are reading data at a much higher rate than writing it by using:

Select one:

- a. a load balancer
- b. database replication
- c. database partitioning



d. caching

Question 14

Not yet answered

Marked out of 1.00

Flag question

What will happen if you try to compile and run this program?

```
public class Inc {
   public static void main(String[] args) {
     int[] i = {1};
     Inc in = new Inc();
     in.increment(i);
                System.out.println(i[i.length - 1]);
        void increment(int[] i) {
   i[i.length - 1]++;
```

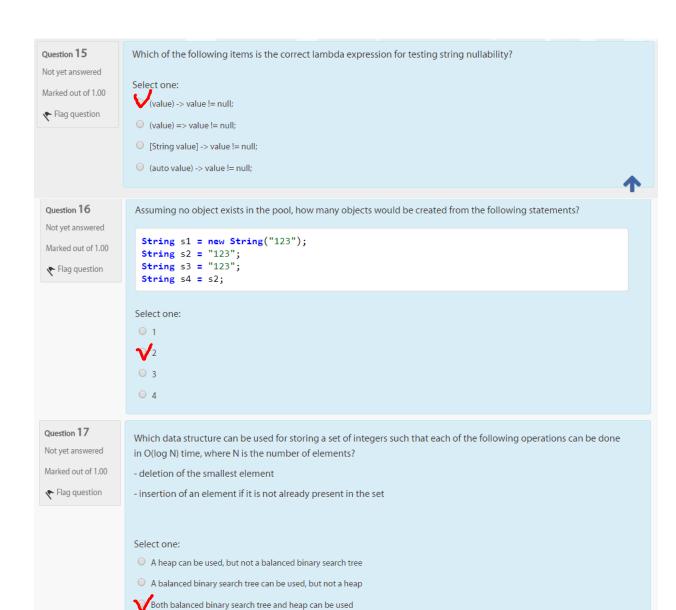
Select one:

Compiler error

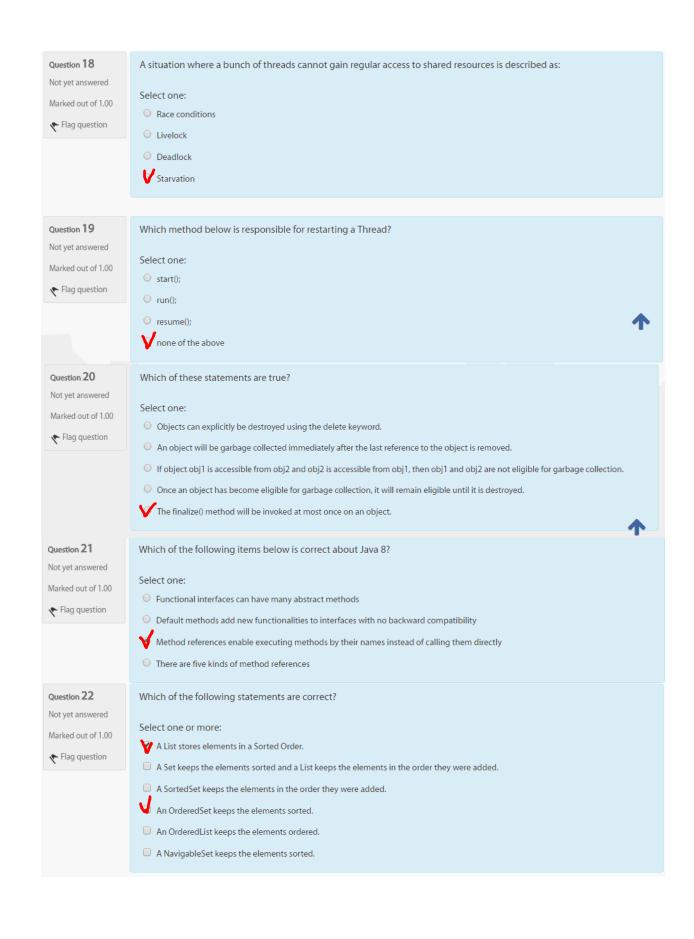


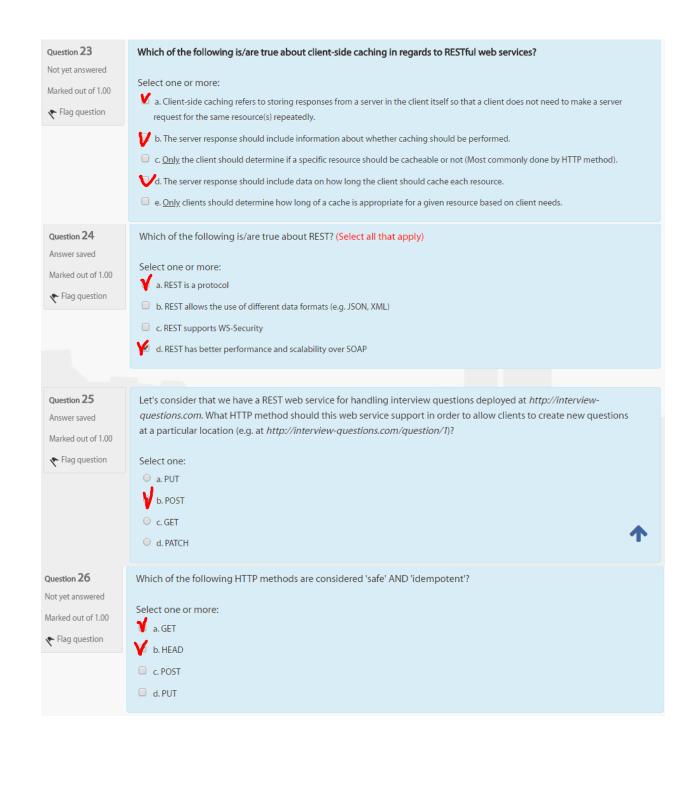
Compiles and runs printing out 2

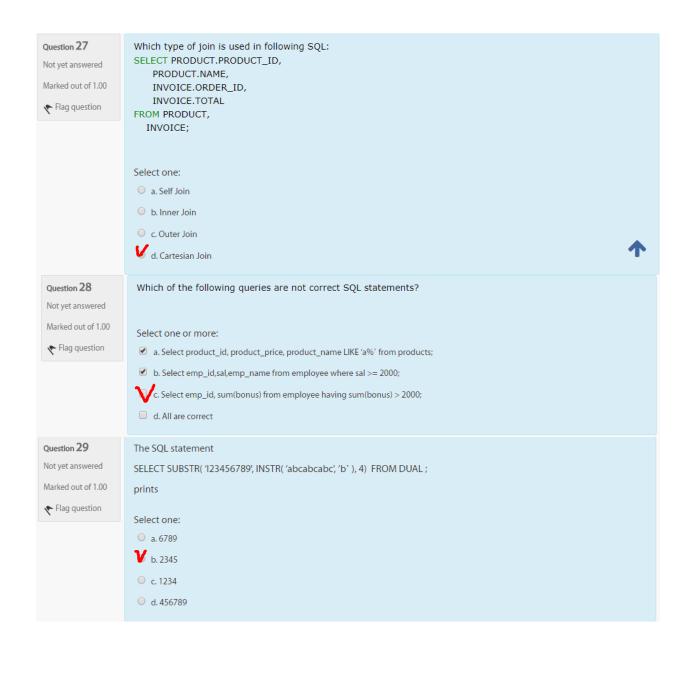
- Ocompiles and runs printing out 1
- An ArrayIndexOutOfBounds Exception at runtime

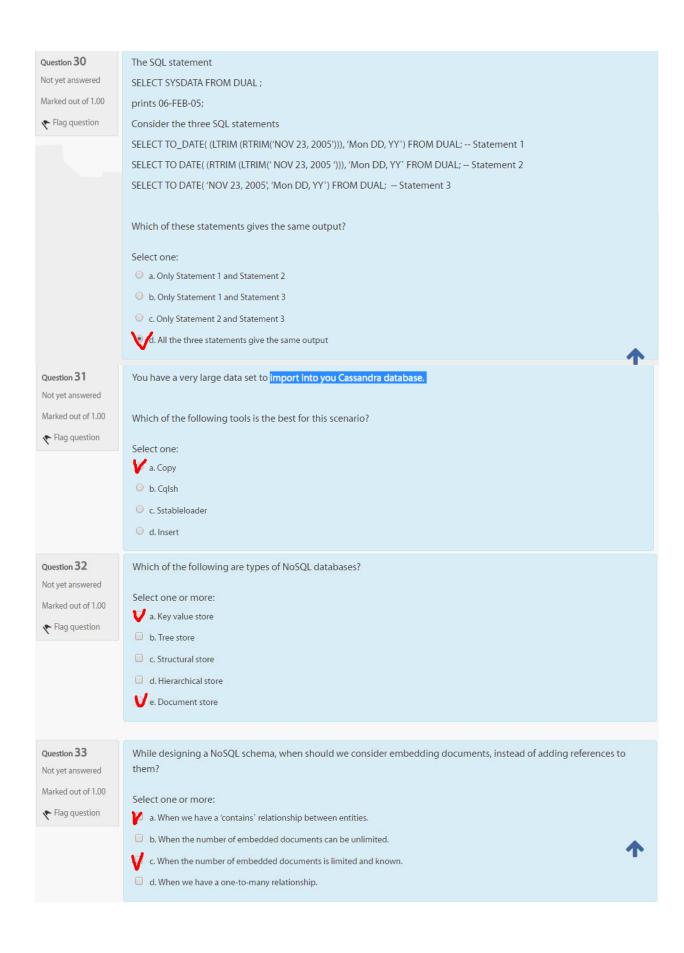


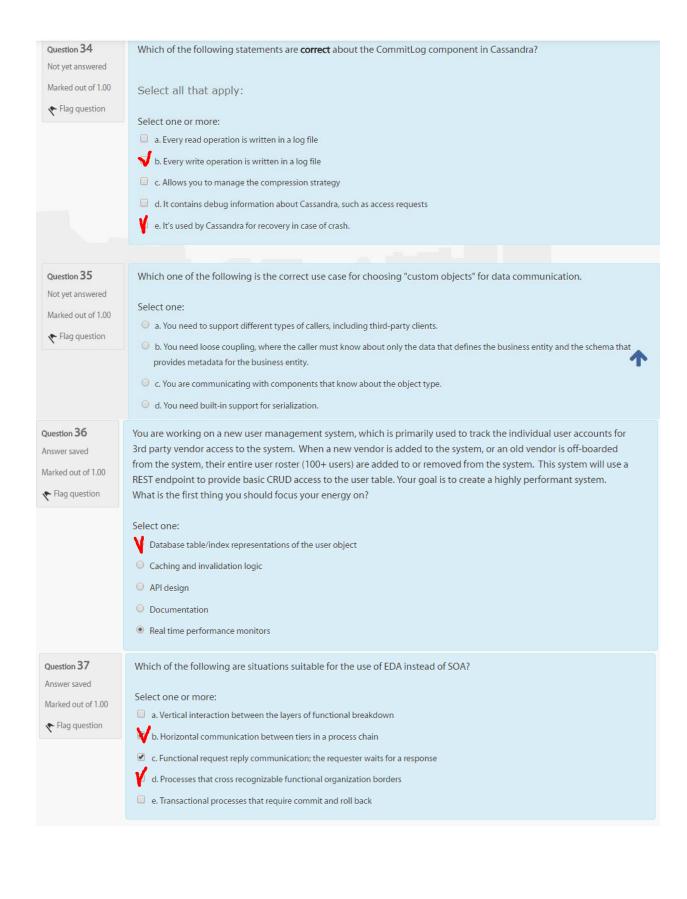
Neither balanced binary search tree nor heap can be used

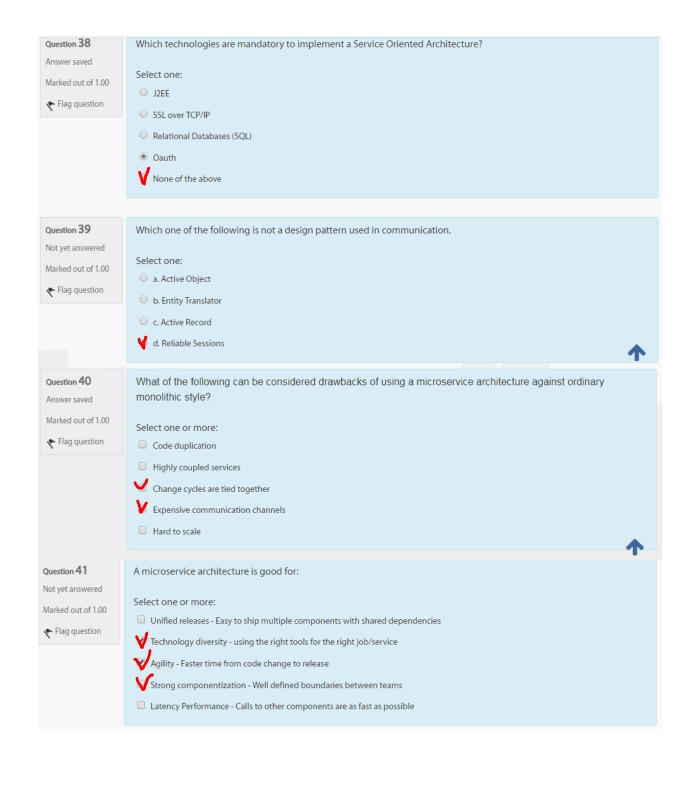


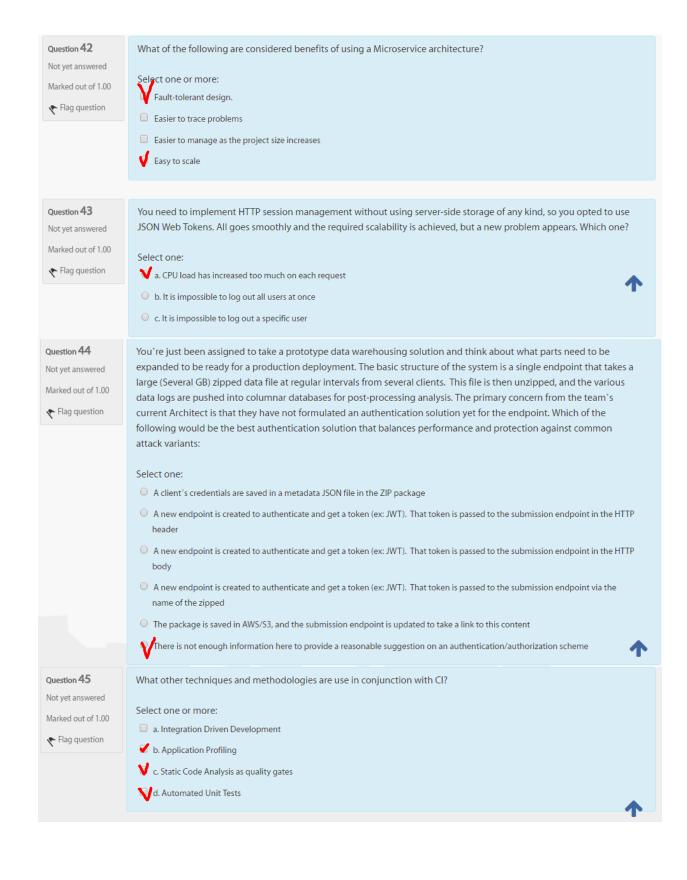


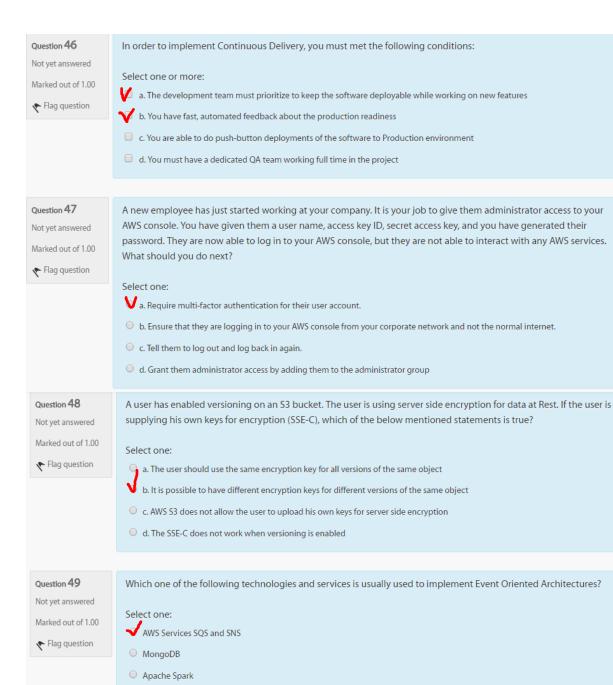






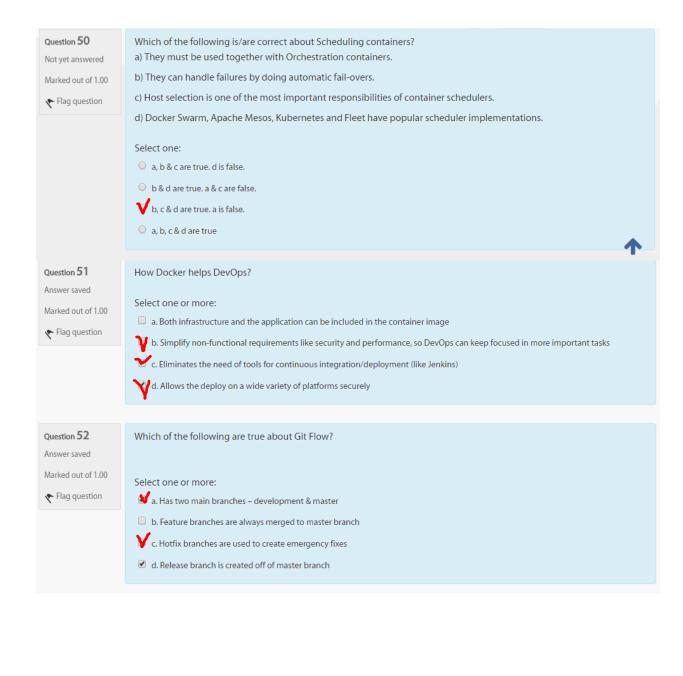


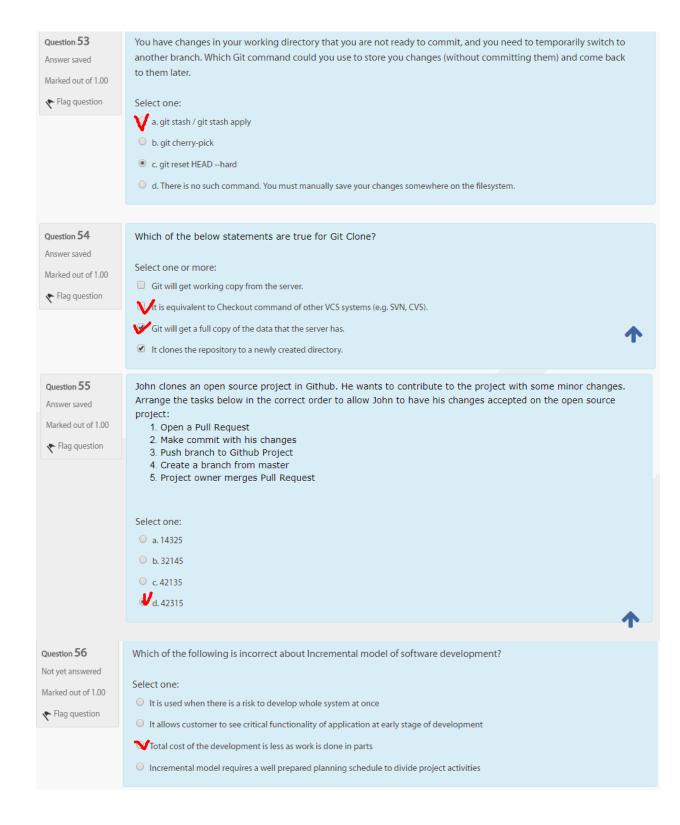


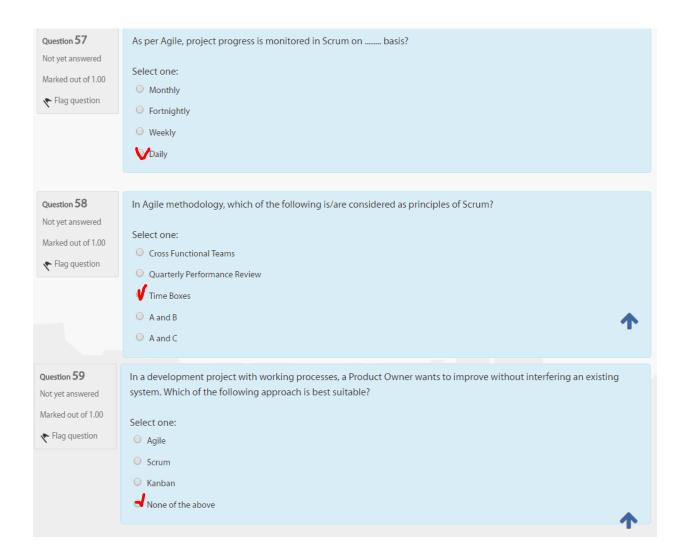


AWS S3

AWS Cloudwatch







Two strings, a and b, are said to be twins only if they can be made equivalent by performing some number of operations on one or both strings. There are two possible operations:

SwapEven: Swap a character at an even-numbered index with a character at another even-numbered index.

SwapOdd: Swap a character at an odd-numbered index with a character at another odd-numbered index.

For example, a = "abcd" and b = "cdab" are twins because we can make them equivalent by performing operations. Alternatively, a = "abcd" and b = "bcda" are not twins (operations do not move characters between odd and even indices), and neither are a = "abc" and b = "ab" (no amount of operations will insert a c into string b).

Complete the code in the editor below, the twins() method with the following params.

An array of n strings named a. An array of n strings named b. The function must return an array of strings where each index i (0 = i < n) contains the string Yes if ai and bi are twins or the string No if they are not.

Input Format:

The internal test cases will pass the following input to the program using the STDIN and expect output on the STDOUT:

The first line contains an integer, n, denoting the number of elements in a.

Each line i of the n subsequent lines (where 0 = i < n) contains a string describing ai.

The next line contains an integer, n, denoting the number of elements in b.

Each line i of the n subsequent lines (where 0 = i < n) contains a string describing bi.

Constraints:

1 = n = 103

```
1 = length(ai), length(bi) = 100
ai and bi are not guaranteed to have the same length.
Strings ai and bi contain lowercase letters only (i.e., a through z).
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;
public class Tester {
    /**
    * Complete the function below.
    * DO NOT MODIFY anything outside this method.
    */
    static boolean[] twins(String[] a, String[] b) {
```

```
boolean[] result = new boolean[a.length];
  return result;
}
* DO NOT MODIFY THIS METHOD!
*/
public static void main(String[] args) throws IOException {
  Scanner in = new Scanner(System.in);
  int n = Integer.parseInt(in.nextLine().trim());
  String[] a = new String[n];
  for(int i = 0; i < n; i++) {
    a[i] = in.nextLine();
  }
  int m = Integer.parseInt(in.nextLine().trim());
  String[] b = new String[m];
  for(int i = 0; i < m; i++) {
    b[i] = in.nextLine();
  }
  // call twins function
  boolean[] results = twins(a, b);
  for(int i = 0; i < results.length; i++) {</pre>
    System.out.println(results[i]? "Yes": "No");
  }
}
```

}