Shrey Arora

Java Code:

1) Java ArrayList

Input Format

The first line has an integer . In each of the next lines there will be an integer denoting number of integers on that line and then there will be space-separated integers. In the next line there will be an integer denoting number of queries. Each query will consist of two integers and .

Constraints

1<=n<=20000

0<=d<=50000

1<=q<=1000

1<=x<=n

Each number will fit in signed integer.

Total number of integers in lines will not cross 100000.

Output Format

In each line, output the number located in position of line. If there is no such position, just print "ERROR!"

Sample Input

5

5 41 77 74 22 44

1 12

4 37 34 36 52

0

3 20 22 33

5

1 3

3 4

3 1

4 3

5 5

Sample Output

74

52

37

ERROR!

ERROR!

Solution:

```
package Assignment102;

import java.util.ArrayList;
import java.util.Scanner;

public class QuesOne {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);

        int n = s.nextInt();
        ArrayList<ArrayList<Integer> > arr =new
ArrayList<ArrayList<Integer>>(n);
        for(int i=0;i<n;i++) {
            int num = s.nextInt();

            ArrayList<Integer> a = new
ArrayList<Integer>();
            for(int k=0;k<num;k++) {
                a.add(s.nextInt());
            }
            arr.add(a);
        }
        System.out.println("The output is");
```

```
        int q = s.nextInt();
        for(int i=0;i<q;i++) {
            int x = s.nextInt()-1;
            int y = s.nextInt()-1;

            if(y>(arr.get(x).size()) - 1) {
                System.out.println("ERROR!");
            }
            else {

System.out.println(arr.get(x).get(y));
            }
        }
    }
}
```

2) Java List

Input Format

The first line contains an integer, (the initial number of elements in).

The second line contains space-separated integers describing .

The third line contains an integer, (the number of queries).

The subsequent lines describe the queries, and each query is described over two lines:

If the first line of a query contains the String Insert, then the second line contains two space separated integers , and the value must be inserted into at index .

If the first line of a query contains the String Delete, then the second line contains index , whose element must be deleted from .

Constraints

1<=N<=4000

1<=Q<=4000

Each element in is a 32-bit integer.

Output Format

Print the updated list as a single line of space-separated integers.

Sample Input

5

12 0 1 78 12

2

Insert

5 23

Delete

0

Sample Output

0 1 78 12 23

SOLUTION:

3) Java Map

You are given a phone book that consists of people's names and their phone number. After that you will be given some person's name as query. For each query, print the phone number of that person.

Input Format

The first line will have an integer denoting the number of entries in the phone book. Each entry consists of two lines: a name and the corresponding phone number.

After these, there will be some queries. Each query will contain a person's name. Read the queries until end-of-file.

Constraints:

A person's name consists of only lower-case English letters and it may be in the format 'first-name last-name' or in the format 'first-name'. Each phone number has exactly 8 digits without any leading zeros.

1<=n<=100000

1<=Query<=100000

Output Format

For each case, print "Not found" if the person has no entry in the phone book. Otherwise, print the person's name and phone number. See sample output for the exact format.

To make the problem easier, we provided a portion of the code in the editor. You can either complete that code or write completely on your own.

Sample Input

3

uncle sam

99912222

tom

11122222

harry

12299933

uncle sam

uncle tom

harry

Sample Output

uncle sam=99912222

Not found

harry=12299933

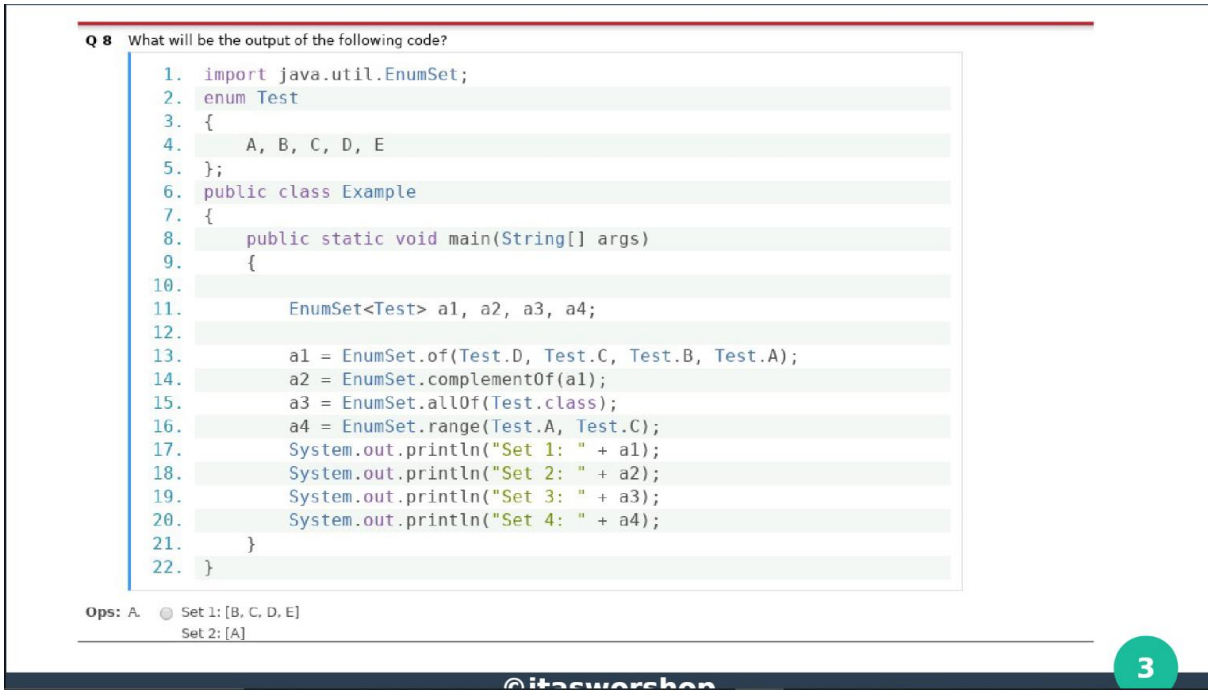
SOLUTION:

```
package Assignment102;

import java.util.HashMap;
import java.util.Scanner;

public class QuesThree {
    public static void main(String[] args) {
        HashMap<String, Integer> hash = new
HashMap<>();
        Scanner sc = new Scanner(System.in);
        int n=sc.nextInt();
        sc.nextLine();
        for(int i=0;i<n;i++){
            String name=sc.nextLine();
            int phn=sc.nextInt();
            sc.nextLine();
            hash.put(name,phn);
        }
        while(sc.hasNext()){
            String str=sc.nextLine();
            try{
                int out=hash.get(str);
                System.out.println(str+"="+out);
            }catch(Exception e){
                System.out.println("Not found");
            }
        }
    }
}
```

Coding MCQ practice set:

```
class MyClass<T, U>
{
    T t;
    U u;
    MyClass(T t, U u)
    {
        this.t = t;
        this.u = u;
    }
      public void print()
    {
        System.out.println(t);
        System.out.println(u);
    }
}
class Main
{
    public static void main (String[] args)
    {
        MyClass <String, Integer> obj =
            new MyClass<String, Integer>("Aarushi", 15);

        obj.print();
    }
}
```


Answer:
Aarushi
15



Q 8 What will be the output of the following code?

```
import java.util.EnumSet;
enum Test
{
    A, B, C, D, E
};
public class Example
{
    public static void main(String[] args)
    {

        EnumSet<Test> a1, a2, a3, a4;

        a1 = EnumSet.of(Test.D, Test.C, Test.B, Test.A);
        a2 = EnumSet.complementOf(a1);
        a3 = EnumSet.allOf(Test.class);
        a4 = EnumSet.range(Test.A, Test.C);
        System.out.println("Set 1: " + a1);
        System.out.println("Set 2: " + a2);
        System.out.println("Set 3: " + a3);
        System.out.println("Set 4: " + a4);
    }
}
```

Ops: A. Set 1: [B, C, D, E]
Set 2: [A]

Q8:

Set 1: [A, B, C, D]

Set 1: [E]
Set 1: [A, B, C, D, E]
Set 1: [A, B, C]

Q 10 In the below piece of code, method go() will throw an exception. The developer has used three catch blocks. To which of the following class types would the thrown exception belongs and what will be the output of the code?

```
1.  class MainClass {
2.   public static void main(String[] args) {
3.    try {
4.     go();
5.    } catch (Exception e) {
6.     System.out.println("1");
7.    } catch (Error e){
8.     System.out.println("2");
9.    } catch (Throwable t){
10.    System.out.println("3");
11.   }
12.  }
13.  static void go(){
14.   go();
15.  }
16. }
```

Ops:
A. Error, 2
B. Compilation Error
C. Throwable, 3
D. Exception, 1

Q10: Option A – Error, 2

Q 14 There are 5 threads in the waiting pool of a monitor 'mon' and all these threads have the same priority. One of the threads is thread1. How can you notify thread1, so that it alone moves from Waiting state to Ready state?

Ops:
A. Execute thread1.notify(); from any code(synchronized or not) of any objects
B. You cannot specify which thread will get notified
C. Execute thread1.notify(); from a synchronized code of any objects
D. Execute mon.notify(thread1); from a synchronized code of any objects

reset answer

Q 15 What should be done if an event listener has to perform a lengthy task i.e. checking spelling in a large document and would not be able to process additional UI events till the task is completed, that makes the program appear to freeze?

Ops:
A. The event listener should perform the UI events on the foremost priority
B. The event listener should hand off long tasks to another thread
C. The event listener should cancel the lengthy task if it is not complete in a specific time
D. The event listener should prioritize the tasks

Q14 Answer : B – You cannot specify which thread will get notified

Q 17 What will be the output of the following code?

```
1.  class TestJoinMethod2 extends Thread{
2.   public void run(){
3.    for(int i=1;i<=3;i++){
4.     try{
5.      Thread.sleep(500);
6.     }catch(Exception e){System.out.println(e);}
7.    System.out.print(i);
8.    }
9.   }
10. public static void main(String args[]){
11.  TestJoinMethod2 t1=new TestJoinMethod2();
12.  TestJoinMethod2 t2=new TestJoinMethod2();
13.  TestJoinMethod2 t3=new TestJoinMethod2();
14.  t1.start();
15.  try{
16.   t1.join(1500);
17.  }catch(Exception e){System.out.print(e);}
18.
19.  t2.start();
20.  t3.start();
21.  }
22. }
```


Q17:

1
2
3
1
1
2
2
3
3

Q 16 What will be the output of the following code?

```
1.
2.  public class Test implements Runnable{
3.  public static void main(String[] args){
4.  Thread t = new Thread(this);
5.  t.start();
6.  }
7.
8.  public void run(){
9.  System.out.println("test");
10. }
11. }
```

Ops:
A. ○ None of the mentioned options
B. ○ The program compiles fine, but it does not print anything because t does not invoke the run() method
C. ○ The program does not compile because this cannot be referenced in a static method.
D. ◉ The program compiles and runs fine and displays test on the console.

reset answer

Q16: Option C – does not compile as it cannot be referenced in static method

Q 18 While performing the basic steps in JDBC for which of the following tasks, you have to supply Oracle driver-specific information in order to allow your program to use the JDBC application programming interface (API) to access a database?

Ops: A. Running a Query and Retrieving a ResultSet Object
B. Creating a Statement Object
C. Closing the ResultSet and Statement Objects
D. Processing the ResultSet Object

Q 19 Which of the following are the correct ways to handle database wamings in JDBC?

1.

```
1.
2.      //Retrieving warning from connection object
3.      SQLWarning warning = conn.getWarnings();
4.      //Retrieving next warning from warning object itself
5.      SQLWarning nextWarning = warning.getNextWarning();
6.      //Clear all warnings reported for this Connection object.
7.      conn.clearWarnings();
```

2.

```
1.
2.      //Retrieving warning from statement object
3.      stmt.getWarnings();
4.      //Retrieving next warning from warning object itself
```


Q18:

Q19:

Q 24 For a web application, if you want to define the scope of an object so that a user's interaction with the web application would be across multiple HTTP requests then which of the following scope you should assign?

Ops: A. @SessionScoped
B. @ApplicationScoped
C. @RequestScoped
D. @ConversationScoped

reset answer

Q 25 Which of the following statements correctly states the difference between a configuration (such as the CLDC) and a profile (such as the MIDP)?

Ops: A. Profile actually uses the vertical set of classes that a configuration defines.
B. Configuration provides a minimum set of class library for the far-ranging devices, and a profile provides the set of APIs for a particular group of devices.
C. Profile is the base of a configuration.
D. Configuration helps to define a minimum set of class libraries for a board range of devices, and profile helps to define a set of APIs for a specific family of devices

Q 22 Suppose an application has a servlet to perfom 'addition operation' as shown below and sets the value in context scope and prints the result.

```
1.
2.  public void doGet(HttpServletRequest req, HttpServletResponse resp) {
3.   PrintWriter out = response.getWriter();
4.   param_1 = request.getParameter("Param1");
5.   param_2 = request.getParameter("Param2");
6.
7.   getServletContext.setAttribute("result",(param_1 + param_2));
8.   out.print("Result =="+getServletContext.getAttribute("result"));
9.  }
```

But it provides wrong output sometimes. In which of the following ways can we solve this issue?

1. By Putting the 'result' attribute in request scope.
2. By Putting the 'result' attribute in session scope.
3. By making the servletContext Synchronized

Ops: A. Only 1 and 3

B. None of the mentioned options can be used to solve this issue

C. Only 2 and 3

D. Only 1

reset answer

Q22: answer C – only 2 & 3