OOP Java

Team overview

Name: Nguyen Hoang Nam

Matriculation number: 1403596

Name: Joanna Galaszewska

Matriculation number: 1344943

Week exercise

Week 1:

Task 1: 100%
Task 2: 100%
Task 3: 100%
Task 4: 100%

Week 2:

Task 1: 100%
Task 2: 100%

Week 3:

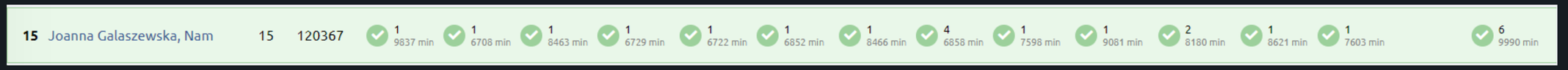
Task 1: 100%
Task 2: 100%
Task 4: 100%
Task 5: 95%

Week 4:

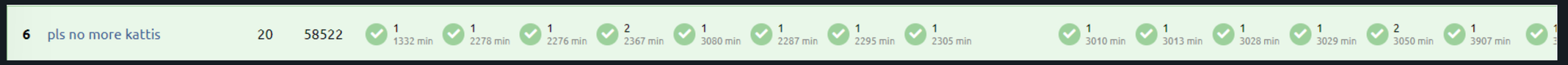
Task 1: 100%
Task 2: 100%
Task 4: 100%
Task 5: 100%

Four weekly contests

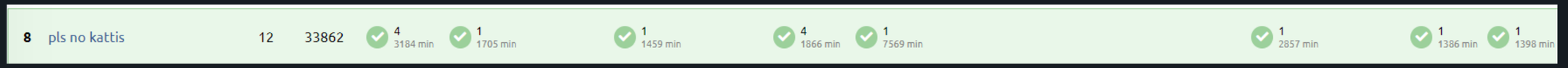
Week 1: 15/21 Rank 15



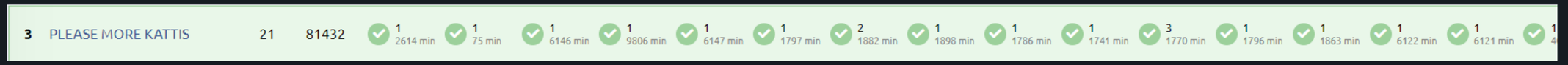
Week 2: 20/21 Rank 6



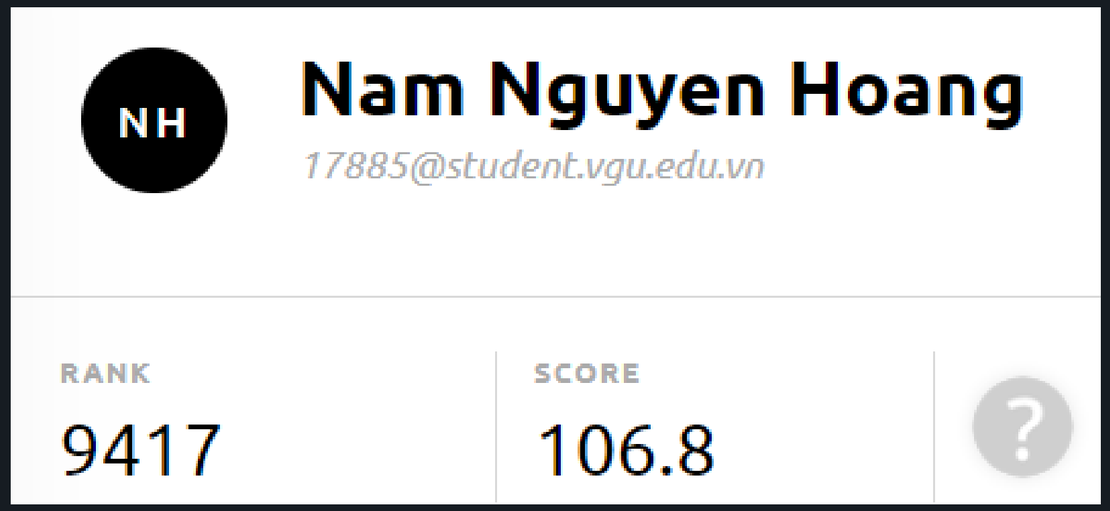
Week 3: 12/24 Rank 8

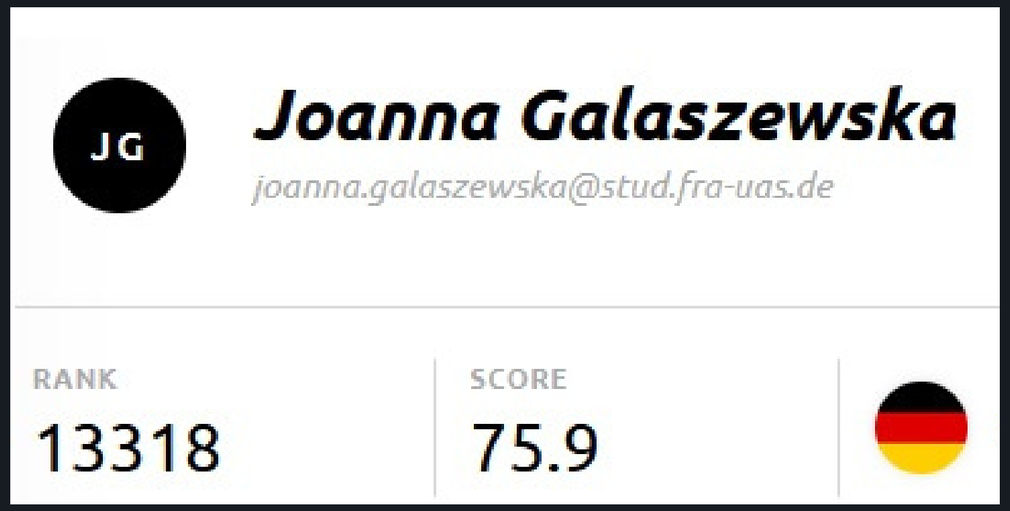


Week 4: 21/21 Rank 3



Kattis profile





Name: Nguyen Hoang Nam

Presentation:

+ Detailed Differences

+ ICPC Awards

Name: Joanna Galaszewska

Don't present anything but

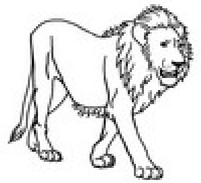
every time there.

Kattis Problem

Week 4: Un-bear-able Zoo

Questiom about creating a list with unique Key und they Value

Un-bear-able Zoo



In his free time, when he's not busy hacking computers, Dr. Back runs a zoo. Every morning he gets up and makes sure that none of the animals have escaped. He has a huge list of all the animals and checks each animal off as he sees it, but thinks this is really inefficient. He only cares about what animal they are, since all similar animals share a cage. So, if he has a white tiger and a siberian tiger, Dr. Back only wants to know that he has **2** tigers.

Given an integer n, and n lines describing animals, output in alphabetical order the animals Dr. Back has in his zoo, followed by their count.

Input

The input will contain multiple test cases, up to **5**. Each test case contains a line containing a single integer n ($0 \leq n \leq 10^3$), followed by n lines of animals with at least one word on every line. An animal name may consist of multiple lowercase or uppercase words, with the last one describing the kind of animal. Animal names may contain apostrophes, hyphens, and periods; e.g., *St. Vincent's Agouti* would be a valid animal name. The input is terminated when n is **0**.

Output

For each test case, output the list number, followed by the animals Dr. Back has in his zoo in lowercase and alphabetical order, with each animal followed by one space and the delimiter | and then another space and their count.

Kattis Problem

Week 4: Un-bear-able Zoo

Questiom about creating a list with unique Key und they Value

Output

For each test case, output the list number, followed by the animals Dr. Back has in his zoo in lowercase and alphabetical order, with each animal followed by one space and the delimiter | and then another space and their count.

Sample Input 1

```
6
African elephant
White tiger
Indian elephant
Siberian tiger
Tiger
Panda bear
1
Blue Russian Penguin
0
```

Sample Output 1

```
List 1:
bear | 1
elephant | 2
tiger | 3
List 2:
penguin | 1
```

```java

    public static void main (String []args) {

        Scanner sc= new Scanner (System.in);

        int n=Integer.valueOf(sc.nextLine());



        int listnumber=1;
        while (n!=0) {


            HashMap<String,Integer> animalsMap=new HashMap<>();
            List <String> animals= new ArrayList<>();

            for (int counter=0;counter<n;counter++) {

                String animal=sc.nextLine();
                String[] parts=animal.trim().split(" ");            //remove places vor und after the words

                String animalType=parts[parts.length-1].toLowerCase();  //last part will be change to lower case
                if(animalsMap.containsKey(animalType)) {                //we check if the key is in a hashmap
                    int value= animalsMap.get(animalType);              //we check the existing value
                    animalsMap.put(animalType, value+1);                //we increment a value with 1

                }else {                                                 // there is no key in the Hashmap
                    animalsMap.put(animalType, 1);                      //We set the value for 1
                    animals.add(animalType);

                }

            }

            Collections.sort(animals);                                  // we sort the list
            System.out.println("List "+ listnumber+":");
            for (int counter=0;counter<animals.size();counter++) {
                System.out.println(animals.get(counter)+" | "+ animalsMap.get(animals.get(counter)));
            }
```

ArrayList HashMap

Problem L
ICPC Awards

The ACM International Collegiate Programming Contest has been held in Vietnam for more than 10 years. The contest is a great chance for the students to meet new friends, broaden their knowledge and of course, win prizes.

Every years, universities can send one or multiple teams to the contest and all universities hope to win prizes. The organizers want to define a rule to award the excellent contestants.

The contest director decided to follow the World Finals policy by having 4 first prizes, 4 second prizes and 4 third prizes. 12 winners out of more than a hundred teams is also a good proportion to recognize the best students.

Since universities can send multiple teams, we don't want one university to swept all the awards. Thus, only the top team from a university can be awarded. It seems harsh for the second best team from one university but do not worry, they will still receive relevant certificates.

The table below is the result of top 10 of Nha Trang Regional Contest 2016. The 4-th (team WINDOWS) and 8-th place (team UBUNTU) did not receive prizes because they were not the top team from University of Engineering and Technology - VNU. Team Metis and team BK.DeepMind are in the same situation.

Input

- The input starts with the number of teams N ($12 \leq N \leq 200$).
- The i-th line of the next N lines contains information about the team that ranks i: the university name and the team name separated by a single space. Both names consists of digits, lowercase and uppercase English alphabet letters only. Both names does not exceed 20 letters in length.
- It is guaranteed that there are at least 12 different universities.

Output

The output should contain 12 lines describing 12 winners. In each line, you should print the university name and the team name separated by a single space. The winners should be listed in the same order as the input.

```
30
Seoul ACGTeam
VNU LINUX
SJTU Mjolnir
VNU WINDOWS
NTU PECaveros
HUST BKJuniors
HCMUS HCMUSSerendipity
VNU UBUNTU
SJTU Metis
HUST BKDeepMind
HUST BKTornado
HCMUS HCMUSLattis
NUS Tourism
VNU DOS
HCMUS HCMUSTheCows
VNU ANDROID
HCMUS HCMUSPacman
HCMUS HCMUSGeomecry
UIndonesia DioramaBintang
VNU SOLARIS
UIndonesia UIChan
FPT ACceptable
HUST BKIT
PTIT Miners
PSA PSA
DaNangUT BDTTNeverGiveUp
VNU UNIXBSD
```

```
Seoul ACGTeam
VNU LINUX
SJTU Mjolnir
NTU PECaveros
HUST BKJuniors
HCMUS HCMUSSerendipity
NUS Tourism
UIndonesia DioramaBintang
FPT ACceptable
PTIT Miners
PSA PSA
DaNangUT BDTTNeverGiveUp
```

```java
public class ICPC {
    Run | Debug
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int n = sc.nextInt();
        sc.nextLine();

        int limit = 0;

        HashSet<String> value = new HashSet<>();
        //ArrayList<String> value = new ArrayList<String>();

        for (int i = 0; i < n; i++) {
            String line = sc.nextLine();
            if (value.add(line.split(regex: " ")[0]) && ++limit <= 12) {
                System.out.println(line);
            }
        }
        sc.close();
    }
}
```

PROBLEMS 19 OUTPUT DEBUG CONSOLE TERMINAL

```
HCMUS HCMUSSerendipity
NUS Tourism
UIndonesia DioramaBintang
FPT ACceptable
PTIT Miners
PSA PSA
DaNangUT BDTTNeverGiveUp
VNU UNIXBSD
CanTho CTUA2LTT
Soongsil Team10deung
Soongsil BezzerBeater
PS C:\Users\namng\Desktop\Java>
```

```java
public class ICPC {
    Run | Debug
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int n = sc.nextInt();
        sc.nextLine();

        int limit = 0;

        //HashSet<String> value = new HashSet<>();
        ArrayList<String> value = new ArrayList<String>();

        for (int i = 0; i < n; i++) {
            String line = sc.nextLine();
            if (value.add(line.split(regex: " ")[0]) && ++limit <= 12) {
                System.out.println(line);
            }
        }
        sc.close();
    }
}
```

PROBLEMS 19 OUTPUT DEBUG CONSOLE TERMINAL

```
HUST BKTornado
HCMUS HCMUSLattis
HCMUS HCMUSTheCows
VNU ANDROID
HCMUS HCMUSPacman
HCMUS HCMUSGeomecry
UIndonesia DioramaBintang
VNU SOLARIS
UIndonesia UIChan
FPT ACceptable
HUST BKIT
PTIT Miners
PSA PSA
DaNangUT BDTTNeverGiveUp
VNU UNIXBSD
```

ArrayList vs HashMap

ArrayList	HashMap
ArrayList implements the List interface.	HashMap implements the Map interface.
ArrayList implements the List interface.	HashMap implements the Map interface.
ArrayList stores element's value and maintains the indexes for each element.	HashMap stores elements **key** & value pair. For each value, there must be a key associated with HashMap.
ArrayList stores only a single object.	HashMap stores elements in Key and value pairs.
We get the element by specifying the index of it in ArrayList.	The elements are being fetched by the corresponding Key in HashMap.
The ArrayList maintains the order of the objects they are inserted.	HashMap does not provide a guarantee of the order in which they are inserted.
ArrayList allows duplicate elements.	HashMap allows duplicate values but does not allow duplicate keys.
The ArrayList always gives **O(1) performance** in best case or worst-case time complexity.	The HashMap get() method has **O(1)** time complexity in the best case and **O(n)** time complexity in worst case.
ArrayList has any number of null elements.	HashMap allows only one null Key and lots of null values.
ArrayList is the index-based data structure supported by the array.	While HashMap is a mapped data structure that works on hashing to obtain stored values.

Kattis Problem

Week 2 - Quick Estimates

This question have problems about logical in answers

Hide

Problem J
Quick Estimates

Let's face it... you are not that handy. When you need to make a major home repair, you often need to hire someone to help. When they come for the first visit, they make an estimate of the cost. Here they must be careful: if they overestimate the cost, it might scare you off, but if they underestimate, the work might not be worth their time.

Photo by Simon A. Eugster

Because the worker is so careful, it can take a long time for them to produce the estimate. But that's frustrating — when you ask for an estimate, you really are asking for the magnitude of the cost. Will this be $10 or $100 or $1 000? That's all you really want to know on a first visit.

Please help the worker make the type of estimate you desire. Write a program that, given the worker's estimate, reports just the magnitude of the cost — the number of digits needed to represent the estimate.

Input

Input begins with a line containing an integer N ($1 \leq N \leq 100$). The next N lines each contain one estimated cost, which is an integer between 0 and 10^{100}. (Some of the workers overcharge quite a bit.)

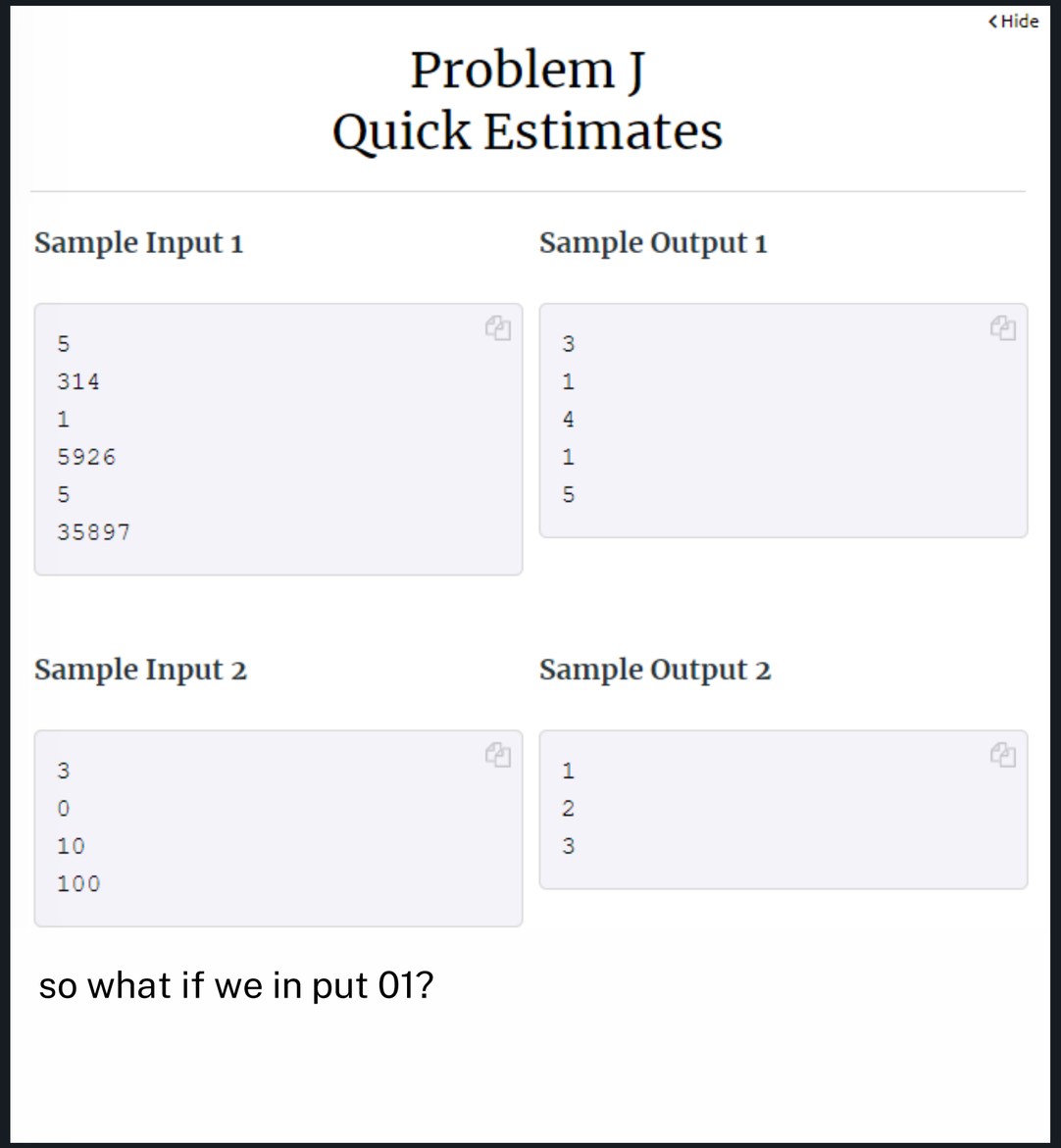
Output

For each estimated cost, output the number of digits required to represent it.

Kattis Problem

Week 2 - Quick Estimates

This question have problems about logical in answers



Hide

Problem J
Quick Estimates

Sample Input 1

```
5
314
1
5926
5
35897
```

Sample Output 1

```
3
1
4
1
5
```

Sample Input 2

```
3
0
10
100
```

Sample Output 2

```
1
2
3
```

so what if we in put 01?

Hide

Problem J
Quick Estimates

Quick_Estimates.java

```java
import java.util.Scanner;
public class quickestimate {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int[] arr = new int[n];
         for(int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
         }
            for(int i = 0; i < n;i++) {
            int length = String.valueOf(arr[i]).length();
            System.out.println(length);
        }
    sc.close();
    }
}
```

Input = 01

Out put = 1 => false

Hide

Problem J
Quick Estimates

Quick_Estimates.java

```java
package Frankfurt.Katis.Week2;

import java.util.Scanner;

public class Quick_Estimates {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int cases = Integer.parseInt(sc.nextLine());

        for (int i = 0; i < cases; i++) {
            System.out.println(sc.nextLine().length());
        }
        sc.close();
    }
}

```

Input = 01

Out put = 2 => true

7 facts about JAVA language you probably did not know

Source: https://www.indiatoday.in/education-today/gk-current-affairs/story/facts-about-java-978690-2017-05-23

7 facts about JAVA language you probably did not know

1. Java was called Oak at the beginning

The original name for Java was Oak. It was eventually changed to Java by Sun's marketing department when Sun lawyers found that there was already a computer company registered as Oak

Gosling and his gang of programmers went out to the local cafe to discuss names and ended up naming it Java

2. It was just an accident!

James Gosling was working at Sun Labs, around 1992. Gosling and his team was building a set-top box and started by "cleaning up" C++ and wound up with a new language and runtime. Thus, Java or Oak came into being.

7 facts about JAVA language you probably did not know

3. You are paid to learn Java

The median salary of a Java developer is USD 83,975. Yes, it pays to be a Java developer and programmers are milking it. There are about 9 million Java developers in the world. (And you are one of them)

4. Second most popular language

Though many would argue that Java is all time favourite among developers, it is the second most popular programming language after C. Java is ranked second in popularity among programming languages.

7 facts about JAVA language you probably did not know

5. The Duke

The Java mascot, 'The Duke' was created by Joe Palrang. Palrang is the same guy who has worked on the Hollywood blockbuster, Shrek. Duke is celebrated at Oracle. A living, life-size Duke is a popular feature at every JavaOne developer conference. Each year, Oracle releases a new Duke personality.

7 facts about JAVA language you probably did not know

6. Most popular user interface

Currently, about 3 billion mobile phones are working in Java, as well as 125 million TV sets and each Blu-Ray player. This language is continually ranked first in the rankings of software developers as the best choice of programming languages.

7. Final is not final in Java

Final actually has four different meanings in Java:
+ final class: the class cannot be extended
+ final method: the method cannot be overridden
+ final field: the field is a constant
+ final variable: the value of the variable cannot be changed once assigned

Thank you

ZIP file:
https://drive.google.com/drive/folders/1mKfhMw9Q7B5aAeXE40ZTK7Z2jjcxnK9f?usp=share_link

Quizzy: https://drive.google.com/drive/folders/1Z-bKWokXCohTI_4UKBeNVIaxOdzS0UeS?usp=share_link

Kattis: https://github.com/Visionman123/Katis-contest

Exercise: https://github.com/Visionman123/OOPJAVA

Presentation:
https://www.canva.com/design/DAFSF7lyaI8/YK9UzC8cKbgIDdmoCne5Lw/edit