

OFFLINE

JUDGE

# OFFLINE JUDGE

資工一B 陳彥呈

OFFLINE

JUDGE OUTLINE

1. Project 想法&目標
2. 內容&特色
3. 遇到的問題
4. Demo

OFFLINE

# JUDGE PROJECT 想法 & 目標

開始

能讓學生在寫程式作業時有即時的回饋

最初想法

**ONLINE JUDGE?**

問題

用JAVA做ONLINE有難度

結果

**OFFLINE JUDGE!**

OFFLINE

JUDGE

登入介面

The screenshot shows the 'Offline Judge' application window in 'Student Mode'. The interface has a black background with white and red text. At the top, 'OFFLINE' is in a white box and 'JUDGE' is in red. Below this is a 'Student' dropdown menu with an 'OK' button. Further down are two input fields: 'Enter your student id' and 'Enter the problem id', each with an 'OK' button. At the bottom are 'Exit' and 'Solve the problem!' buttons. Two grey callout boxes with arrows point to the input fields: '輸入學生編號' (Enter student ID) points to the first field, and '輸入題號' (Enter problem ID) points to the second field. A third grey box labeled '解題' (Solve problem) points to the 'Solve the problem!' button.

Offline Judge

OFFLINE JUDGE

Student OK

Enter your student id OK

Enter the problem id OK

Exit Solve the problem!

輸入學生編號

輸入題號

解題

學生模式

The screenshot shows the 'Offline Judge' application window in 'Assistant Mode'. The interface is similar to the student mode but with different options. It has 'OFFLINE' in a white box and 'JUDGE' in red at the top. Below is a 'TA' dropdown menu with an 'OK' button. At the bottom are 'view score' and 'Exit' buttons. A grey callout box labeled '觀看成績' (View score) points to the 'view score' button.

Offline Judge

OFFLINE JUDGE

TA OK

view score Exit

觀看成績

助教模式

# OFFLINE

# JUDGE 解題畫面

題目

Offline Judge

**Question**

The description of each test case is given below: Three non-negative integers e, f, c, e equals the number of empty soda bottles in Tim's possession at the start of the day, f is the number of empty soda bottles found during the day, and c is the number of empty bottles required to buy a new soda. For each test case print how many sodas did Tim drink on his extra thirsty day? Look at the sample output for details. \*\*\*Input end with -1\*\*\*

**Example Input**

9 0 3

**Example Output**

4

**Your Code**

```
#include<iostream>
using namespace std;
int main(){
    int a,b,c;
    while(cin>>a){
        if(a==-1)break;
        cin>>b>>c;
```

Code Submitted!

**AC**

Back to menu Submit

範例輸入

範例輸出

程式碼貼上處

結果

# OFFLINE

# JUDGE 成績檢視畫面

輸入學生編號

1

Now showing student 1's score ...

1.WA	7.null	13.null	19.null
2.AC	8.null	14.null	20.null
3.AC	9.null	15.null	21.null
4.AC	10.null	16.null	22.null
5.AC	11.null	17.null	23.null
6.null	12.null	18.null	24.null

Back

View Score

顯示學生各題作答結果  
正確出現AC  
錯誤出現WA  
未作答則是null

# OFFLINE

## JUDGE 題目格式

--a--

題目

--b--

範例輸入

--c--

範例輸出

--d--

測資

--e--

答案

```
5 - Notepad
File Edit Format View Help

--a--
You are given a string consisting of parentheses 'R' and 'L'. A string of this type is said to be correct
(a) if it is the empty string
(b) if A and B are correct, AB is correct,
(c) if A is correct, RAR and LAL is correct.
Write a program that takes a sequence of strings of this type and check their correctness. Your
program can assume that the maximum string length is 128.
Input is a sequence of strings of parentheses 'R' and 'L', one string
a line.***Test cases end with -1***

--b--
rrlRLRR
--c--
Yes
--d--
rLrRILR
rrrllllLRRL
rRlrrrLrRRLRL
-1
--e--
Yes
No
Yes
```

# JUDGE 結果EXCEL

正確出現AC  
錯誤出現WA  
未作答則是null



OFFLINE

# JUDGE 遇到的問題1：命列列

- 問題:

Java在執行指令方面並不是非常方便

- 解決方法:

使用C++寫出編譯程式，讓JAVA直接執行

OFFLINE

## JUDGE 遇到的問題2：寫入速度

- 問題:

程式碼編譯再加上寫入速度過慢，導致讀到上次結果

- 解決方法:

使用flush()加速寫入，並且在讀結果之前加入延遲

OFFLINE

JUDGE

DEMO