

C++ Test

2022 機器人實驗室 C++測驗

出題者：吳政彥

Outline

Simple

Q1 – GPA Calculator

Q2 – Vocabulary 7000

Medium

Q3 – Pointer

Q4 – Bowling

Usage of testcase

Submission Format

Q1 – GPA Calculator

本學期快結束，Offliner 已經知道自己各科的學期總成績，想知道自己這學期的 GPA 是多少，請各位幫 Offliner 寫一個 C++ 的程式來計算他的 GPA 是多少。

下表為百分制與等第制對照表

| 百分制 | 等第制 |
|----------|----------|
| 90 - 100 | A+ (4.3) |
| 85 - 89 | A (4) |
| 80 - 84 | A- (3.7) |
| 77 - 79 | B+ (3.3) |
| 73 - 76 | B (3) |
| 70 - 72 | B- (2.7) |
| 67 - 69 | C+ (2.3) |
| 63 - 66 | C (2) |
| 60 - 62 | C- (1.7) |
| 0 - 59 | F (0) |

Input

第一個數字 M 為 Offliner 這學期修課數量(M 不會超過 int 上界)，接著會接續 M 行資料，每行資料會有兩個數字，第一個為該科目的百分制分數，第二個為該科目的學分數，這兩個數字之間會用空白隔開，其中該科目若不及格則會顯示 F，若該科目停修則會顯示 W，且停修科目不列入 GPA 計算。科目的分數需先採四捨五入至整數再做 GPA 的權重計算。

Output

輸出 Offliner 該學期的 GPA，該值須四捨五入至小數第二位，若不及格的學分數大於該學期總學分數(含停修學分數)的一半，則需換行印出 flunk out。

Example 1

| Input | Ouput |
|----------------------------------|-------|
| 4 97 3 78 2 84 2 F 3 | 2.69 |

Example 2

| Input | Ouput |
|--------|-------|
| 7 | 2.35 |
| 91 2 | |
| W 1 | |
| 81.5 2 | |
| F 3 | |
| 78 2 | |
| 66.7 3 | |
| 59.5 2 | |

Example 3

| Input | Ouput |
|--------|-------|
| 5 | 1.77 |
| 84.6 2 | |
| F 2 | |
| 81.5 2 | |
| F 3 | |
| 69 1 | |

Example 4

| Input | Ouput |
|----------|-----------|
| 3 | 1.6 |
| F 3 | flunk out |
| 94.215 1 | |
| 83 1 | |

Q2 – Vocabulary 7000

Offliner 在當高中生的英文家教時，遇到了一個的學生名叫小明，他非常不喜歡使用手機軟體來背 7000 單，因為軟體通常會內建許多廣告，這容易使他分心，因此請你使用 **C++**來撰寫能輸入英文單字後，並透過查詢 7000 單的 **csv** 資料表來找到其單字的詞性與中文意思。

7000 單的 **csv** 資料表有兩欄，第一欄為英文單字，第二欄為單字詞性與單字中文意思，單字詞性在前而單字中文意思在後。

Input

首先會先輸入數字 **N**(**N** 不會超過 **int** 上界)，接下來會接續輸入 **N** 個英文單字，英文單字可能會摻雜大小寫。

Output

輸出 **N** 個單字的詞性與中文意思，兩者之間用一個空白隔開，若找不到該單字直接輸出 **Unknown**。

Example 1

| Input | Ouput |
|---------|-------------|
| 4 | v. 吸收；理解 |
| absorb | n. 反派角色 |
| villain | n. 甲狀腺 |
| thyroid | n./v. 汗水；出汗 |
| sweat | |

Example 2

| Input | Ouput |
|---|---------------|
| 8 | n./v. 爭論 |
| Debate | n. 長袍 |
| RoBE | adj. 對...不感興趣 |
| indifFeReNt | Unknown |
| PnEuMonoultramicRoscopiCsilicOVOLcanoconiosis | adj. 理論(上)的 |
| THEORETICAL | n. 愛國者 |
| patRioT | n. 護照 |
| paSSpoRt | Unknown |
| b00k | |

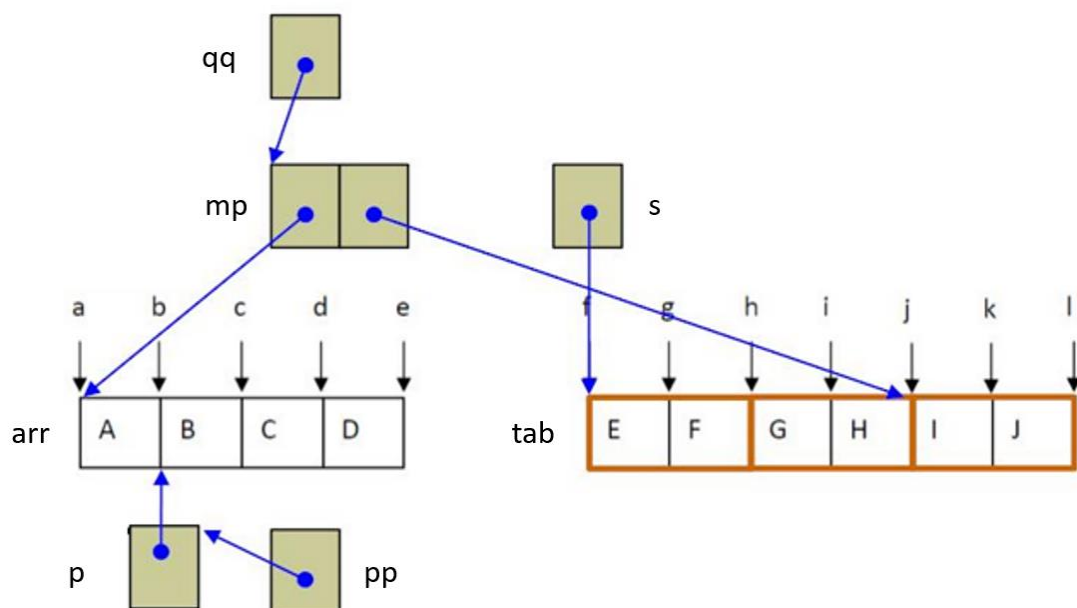
Q3 – Pointer

由於 Offliner 苦手於 C/C++ 的指標，不過他已經畫好該程式的指標圖，也完成表格的一部分，所以請你幫幫他完成剩下的部分吧(跪

Code

```
1. int main()
2. {
3.     int arr[4];
4.     int tab[3][2];
5.     int *p = arr + 1;
6.     int *mp[2] = {arr, tab[2]};
7.     int **pp = &p;
8.     int **qq = mp;
9.     int (*s)[2] = tab;
10. }
```

Pointer Graph



Point table

請完成表格剩下的部分

| Expression | Address or variable | Data type |
|--------------|---------------------|------------|
| arr | {A, B, C, D} | int [4] |
| arr[0] | A | int |
| arr+1 | b | int * |
| tab+1 | h | int (*)[2] |
| tab[1] | {G, H} | int [2] |
| tab[1]+1 | | |
| tab[1][1] | | |
| &tab[1][1] | | |
| &tab[1][1]+1 | | |
| p+0 | | |
| p+1 | | |
| *p | | |
| p[2] | | |
| mp[0][1] | | |
| mp[1][1] | | |
| *pp+2 | | |
| qq[1] | | |
| qq[1]+1 | | |
| qq[0][0] | | |
| qq[1][0] | | |
| s[0][1] | | |
| s[1][0] | | |

※Hint：其實也可以把陣列填滿不同值，再把每個位置的記憶體值記下來，最後直接把未完成的 **Expression** 給印出來看看

Q4 – Bowling

最近因為實驗室有人確診，不幸受到波及的 Offliner 只能乖乖進行 3+4 隔離，但不幸中的大幸是 Nintendo Switch 推出了 Sports 這款遊戲，而 Offliner 深陷在保齡球這運動中



但保齡球的計分方式有些複雜，就由 Offliner 娓娓道來。首先可以看到下圖為保齡球 10 局的計分表

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|----|
| | | | | | | | | | |

每局有兩次的出手機會(第 10 局例外)，假如第一局的第二次出手擊倒 8 瓶則會記為

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|----|
| 8 | | | | | | | | | |

若第一局的第二次出手只擊倒 1 瓶，則會記為

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|----|
| 8 | 1 | | | | | | | | |

接著就可以計算第 1 局的得分為 9，會記在第 1 局的下方

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|----|
| 8 | 1 | | | | | | | | |
| 9 | | | | | | | | | |

接著，如果第 2 局的第一次出手擊倒 7 瓶，而第二次出手把剩下的 3 瓶都擊倒，則稱為 Spare(Spare 會註記/)。發生 Spare 時會將上局的得分加上本局得滿分(10 分)，再加上下一次出手的分數當成獎勵分，所以第 2 局發生 Spare 時會將上一局的 9 分加上第 2 局的滿分 10 分，再加上第 3 局的第一次出手 9 分，最後第 2 局的計分為 28

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|---|---|---|---|---|---|---|----|
| 8 | 1 | 7 | / | | | | | | |
| 9 | 28 | | | | | | | | |

若第 3 局又發生 Spare，會將第二局的 28 分加上本局滿分 10，再加上下一局的第一次出手 7 分，所以第 3 局會註記 45

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|----|---|---|---|---|---|---|----|
| 8 | 1 | 7 | / | 9 | / | 7 | | | |
| 9 | 28 | 45 | | | | | | | |

第 4 局因為沒有把剩下的球瓶都擊倒，所以第 4 局只會由第 3 局的分數加上本局得分 9 分，因此第 4 局為 54 分

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|----|----|---|---|---|---|---|----|
| 8 | 1 | 7 | / | 9 | / | 7 | 2 | | |
| 9 | 28 | 45 | 54 | | | | | | |

而當第 5 局一次出手就全倒，稱為 Strike(Strike 會標記 x)，Strike 可以獲得之後兩次出手的分數，所以第 4 局的 54 分，加上本局滿分 10 分，再加上之後一次出手的 10 分，還有之後第 2 次出手的 7 分，最後第五局記分為 81 分

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|----|----|----|---|---|---|---|----|
| 8 | 1 | 7 | / | 9 | / | 7 | 2 | x | |
| 9 | 28 | 45 | 54 | 81 | | | | | |

第 6 局又發生 **Strike**，分數會由第 5 局的 81 加上第 6 局得滿分 10 分，在加上後兩次出手的得分，分別為 7 分與 2 分，所以第 6 局會註記為 100 分

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|----|----|----|-----|---|---|---|----|
| 8 | 1 | 7 | / | 9 | / | 7 | 2 | x | |
| 9 | 28 | 45 | 54 | 81 | 100 | | | | |

依此類推，就可以算得遊玩圖片中的 164 分。而第 10 局比較特別，最多有 3 次的出手機會，但如果在第 10 局的前兩次出手沒能擊倒所有球瓶則會喪失第三次的出手機會，如

| 10 |
|----|
| 7 |
| 2 |

為了程式方便，只要是該局 **Strike** 不用第 2 次出手，或者第 10 局喪失出手機會都會註記-，如下圖兩種情況

| 10 | 6 |
|----|-----|
| 7 | x |
| 2 | - |
| - | 100 |

既然已經知道保齡球的計分規則，請幫幫 **Offliner** 來計算他的總分與他的積分，**Nintendo Switch Sports** 只要有參加就有 40 積分，當打出一次 **Spare** 會乘上 1.1，而打出 **Strike** 會乘上 1.2，如遊戲畫面中，**Offliner** 共擊出 3 次 **Strike** 與 4 次 **Spare**，所以能獲得 $40 \times 1.2 \times 1.2 \times 1.2 \times 1.1 \times 1.1 \times 1.1 \times 1.1 \div 101$ 積分

Input

會輸入最多 21 個出手標記(第 1 到 9 局各 2 個，第 10 局 3 個，且分別用一個空白隔開)，由於 **Nintendo Switch Sports** 是採 16 人淘汰制，因此不一定會打滿 10 局，也可能輸入低於 21 個出手標記，例如 6 個，表示只打 3 局。

Output

輸出最後總得分與積分，積分採四捨五入至整數

Example 1

| Input | Output |
|---|------------|
| $8 \ 1 \ 7 \ / \ 9 \ / \ 7 \ 2 \ x - x - 7 \ 2 \ x - 9 \ / \ 7 \ / \ 8$ | 164 101 |

Example 2

| Input | Output |
|---|------------|
| $x - x - x - x - x - x - x - x - x - x - x \ x \ x$ | 300 248 |

Example 3

| Input | Output |
|---|------------|
| $x - 9 \ / \ x - 8 \ / \ x - 7 \ / \ x - 6 \ / \ x - 1 \ / \ x$ | 200 192 |

Example 4

| Input | Output |
|-------------------------|----------|
| $8 \ / \ 7 \ / \ 6 \ /$ | 43 53 |

Usage of testcase

在雲端中可以發現有測資，但可以發現這些測資是.in 檔與.out 檔，假設寫完 C++的程式如下

Code

```
1. #include<iostream>
2.
3. using namespace std;
4.
5. int main(void)
6. {
7.     int a, b;
8.
9.     cin >> a >> b;
10.    cout << a + b << endl;
11.
12.    return 0;
13. }
```

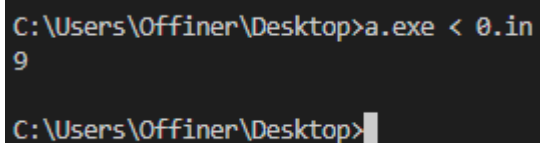
編譯後會產生一個 exe 的執行檔，而有個.in 的測資檔案，內容如下

| |
|------|
| 0.in |
| 4 5 |

假設產生的執行檔是 a.exe，而測資是 0.in，這時只要在終端機執行

| |
|--------------|
| a.exe < 0.in |
|--------------|

就可以看到該執行檔輸出的結果



```
C:\Users\Offiner\Desktop>a.exe < 0.in
9
C:\Users\Offiner\Desktop>
```

再來就可以去對照 0.out 的檔案內容是否與執行結果相同!

Submission Format

繳交格式請如下：

```
<your name>/  
  q1/  
    main.cpp  
  q2/  
    7000.csv  
    main.cpp  
  q3/  
    pointer_table_screen_shot.jpg (png)  
  q4/  
    main.cpp
```

程式部分會透過 Shell Script 去跑測資，所以請照上述格式包好後上傳至繳交的雲端連結

可以的話，蠻建議大家可以用類別的形式來實現，因此有 func.cpp 與 func.h 也直接放入該問題資料夾中即可

由於這些問題是我突發奇想，測資也只是簡單產生，所以可能存在一些 Edge Case 或者 Bug，因此對問題或測資有任何疑問、需要更多測資、想找我討論甚至是覺得題目太簡單沒挑戰性者，歡迎透過 Line 私訊我~