

# HW11-1. 數字加減

請撰寫一個 C++ 程式來計算兩數的總合和差值

請實作以下三個函數：

1. `bigNumberAdd()`

計算兩個整數的總和並回傳

2. `bigNumberSub()`

計算兩個整數的差值並回傳

3. `compareAndSwap()`

比較兩個整數的大小，若(`num1 > num2`) == True 不變

反之則做交換(swap)，確保 `num1 >= num2`

## Input

第一行為非負整數 `num1`，以字串形式讀入

第二行為非負整數 `num2`，以字串形式讀入

## Output

第一行為兩數總和，以 “`sum:`” 作為開頭

第二行為兩數差值，以 “`diff:`” 作為開頭

Sample Input 1

123

456

Sample Output 1

sum: 579

diff: 333

先別急著寫請往下看

「這次作業竟然那麼簡單嗎？」我心裡想著。  
說真的，看到題目那一刻，我幾乎笑了出來。

「只要實作數字加減？」——這是什麼國小程度的題目？

我甚至還心想，這次應該能馬上交作業了吧。  
就在我幻想提早下課、悠閒去買珍奶的時候，一道熟悉又帶點惡意的聲音在我背後響起。

「哎呀，你果然這麼想呢。」  
我回過頭，正對上段助教嘴角那抹彷彿看穿一切的笑容。

學姊今天依舊帶著銀色的耳環，髮尾微微卷起，陽光從窗外照在她的側臉上，讓整個畫面突然閃閃發亮。

「段、段助教，妳今天怎麼會來 206……」  
「來看看有沒有人輕敵呀。」她笑得像是什麼都知道  
「就像我預料的一樣。」

我還沒回過神，她已經走到我的桌旁。  
距離近得誇張，是那種會讓人分不清是心動還是恐懼的距離。

「妳、妳該不會說，這次作業有陷阱吧？」  
「陷阱？」她歪著頭，「不算陷阱喔。只是……一點小惡作劇。」  
「什、什麼惡作…?!」

我話還沒說完，她伸出食指，輕輕抵在我的嘴唇上。  
「你這個小傻瓜，真。可。愛。」

我僵住了，心跳像 CPU 過熱一樣快。  
等等，這是什麼劇本？

她的指尖離開時，我才注意到投影幕亮了起來——新的一頁簡報出現了。  
【第二頁：本題需支援任意長度整數的運算】

# HW11-1. 大數加減 (這才是真的)

和 Python 不同，C++較接近電腦運作本質，重視效能與系統層級操作。因此在程式碼的撰寫上也相對比較嚴謹。一個例子是，C++的 int 以及 long long 都有大小上的限制，超過一定的數值會造成溢位問題而錯誤。這時我們就需要使用字串或陣列的方式來進行大數的運算。

請實作以下三個函數：

1. `bigNumberAdd()`

計算兩個整數的總和並回傳

2. `bigNumberSub()`

計算兩個整數的差值並回傳

3. `compareAndSwap()`

比較兩個整數的大小，若(`num1 > num2`) == True 不變

反之則做交換(swap)，確保 `num1 >= num2`

請使用作業所提供的 template.cpp 來撰寫

\*提示：可以想像以直式的方式對每個位數做計算

## Input

第一行為任意長度的非負整數 `num1`，以字串形式讀入

第二行為任意長度的非負整數 `num2`，以字串形式讀入

## Output

第一行為兩數總和，以 “`sum:`” 作為開頭

第二行為兩數差值，以 “`diff:`” 作為開頭

Sample Input 1	Sample Output 1
123 456	sum: 579 diff: 333

Sample Input 2	Sample Output 2
999999999999999999 111111111111111111	sum: 111111111111111110 diff: 888888888888888888

Sample Input 3	Sample Output 3
123456789012345678901234567890 987654321098765432109876543210	sum: 1111111110111111111011111111100 diff: 864197532086419753208641975320

將 char 轉為 int 做法範例：

```
char ch = '5'; // 字符 '5' · ASCII Code 為 53

// C++ 將char的運算 '5'-'0' 視為int類型的 53-48 = 5
int digit = ch - '0'; // 字符 '0' · ASCII Code 為 48
cout << digit; // 5
```

## HW11-2. 一袋米要扛幾樓

曉組織的佩恩好奇一袋米要扛幾樓，所以他決定自己扛扛看。

他一步能爬一層樓或兩層樓，

也能夠使用技能消耗查克拉一步走三層樓，但用完技能需要冷卻一步。也就是說不能夠連續兩次爬三層樓。

請你算出佩恩有多少種走法能將 n 層樓爬完：

以 4 層樓為例：

- 1, 1, 1, 1
- 1, 1, 2
- 1, 2, 1
- 2, 1, 1
- 2, 2
- 1, 3
- 3, 1

共有 7 種走法

### Input

一行非負整數 n 代表要爬幾層樓， $1 \leq n \leq 30$

Ex. 4

### Output

一行非負整數代表爬完 n 層樓的所有走法數量

Ex. 7

Sample Input 1

4	7
---	---

Sample Output 1

Sample Input 2

5	13
---	----

Sample Output 2

Sample Input 3

6	23
---	----

Sample Output 3

Sample Input 4

30	36555476
----	----------

Sample Output 4



## 繳交格式

1. 上傳內容須為.cpp 檔
  - 檔名為：班級 XX-X-學號  
(XX-X:第幾份作業, i.e. A01-1-114502000)
2. 請針對程式內的重要功能進行「適當註解」，簡單說明該段程式碼的作用，註解的部分會納入評分考量，請勿隨意註解
3. 記得在 Online Judge 系統上測試，若在正常繳交期限前未有測試紀錄，將扣該次作業 10 分
4. 若檔名錯誤，扣該次作業 5 分

## 繳交期限

- **2025/11/21 23:59** 前將檔案上傳到 ee-class 上

## 遲交期限

- **2025/11/23 23:59** 前將檔案上傳到 ee-class 上
- 該次作業分數打八折
- 遲交期限過後不再開放繳交作業，該次以 0 分計算