

# 國立鳳新高中 112 年資訊學科競賽校內賽

## 注意事項

1. 本次競賽總共有7題，使用ZeroJudge自動評分，請上傳原始程式碼(C/C++)
2. 考試時間共150分鐘，競賽結束後將無法再上傳程式碼，比賽期間請勿交談討論。
3. 請勿在程式中加入 `system("pause")` 等指令。
4. 每次上傳之間必須間隔60秒，每題以最佳評分結果計分。

## 各題概況

| 題目名稱 | 時間限制 | 記憶體限制    | 評分方式            |
|------|------|----------|-----------------|
| 美味布丁 | 0.1s | 64MBytes | Tolerant (寬鬆比對) |
| 電梯巨星 | 1s   | 64MBytes | Tolerant (寬鬆比對) |
| 資料驗證 | 1s   | 64MBytes | Strictly (嚴格比對) |
| 市集採買 | 0.3s | 64MBytes | Tolerant (寬鬆比對) |
|      |      |          |                 |
| 滿頭問號 | 1s   | 64MBytes | Special (自訂比對)  |
|      |      |          |                 |

## A. 美味布丁

日頭赤炎炎，在山頭奔波的烏骨雞倍感炎熱，感覺都快中暑了！「好想吃一口沁涼的布丁！」烏骨雞說。因此，在山腳的巨星廚師雞決定做一道簡單的布丁來滿足烏骨雞。廚師雞找到的布丁食譜如下：

「

鮮奶280克

雞蛋液120克

鮮奶油30克

砂糖30克

焦糖5克

」

雖然廚師雞做的布丁超級完美，但牠還是持續更新食譜比例，追求極致美味的布丁。礙於食材有限，廚師雞想盡可能用現有的食材，做多一點布丁，但牠做的新食譜太多，沒辦法一個一個算，因此想用程式來算出牠最多可以做出多少克的布丁。

以牠找到的食譜為例，化為最簡整數比的食譜比例為鮮奶：雞蛋液：鮮奶油：砂糖：焦糖=56：24：6：6：1，假設牠今天有鮮奶336克、雞蛋液144克、鮮奶油36克、砂糖36克、焦糖5克，那牠最多能做出465克的布丁。

今天廚師雞依序給你5個食材的比例，還有5個食材現在有的克數，其中因為食材一定是最新鮮的，所以食材現有的克數一定是食材比例的正整數倍，請你幫牠寫一個程式，算出牠最多可以做出多少克的布丁。

## 輸入格式

第一行有五個正整數，分別代表鮮奶、雞蛋液、鮮奶油、砂糖、焦糖的比例，第二行有五個正整數，分別代表鮮奶、雞蛋液、鮮奶油、砂糖、焦糖的現有克數。其中每個數值都不超過100000，且食材現有的克數一定是食材比例的正整數倍。

## 輸出格式

輸出只有一個數值，代表最多能做出的布丁克數。

## 測試資料

|   |              |
|---|--------------|
| 輸入範例1<br>56 24 6 6 1<br>336 144 36 36 5 | 輸出範例1<br>465 |
| 輸入範例2<br>23 10 1 5 6<br>46 50 3 80 30   | 輸出範例2<br>90  |

## 測試資料說明

範例測資1：在這個範例中，只有焦糖為食材比例的5倍，其餘食材皆為食材比例的6倍，因此焦糖是最快耗盡的，故整體比例為鮮奶：雞蛋液：鮮奶油：砂糖：焦糖=280：120：30：30：5。最多做出465克的布丁。

範例測資2：在這個範例中，鮮奶為最快耗盡的，整體比例為鮮奶：雞蛋液：鮮奶油：砂糖：焦糖=46：20：2：10：12。最多做出90克的布丁。

## 配分

|        |          |      |                 |
|--------|----------|------|-----------------|
| 記憶體限制  | 64MBytes | 評分方式 | Tolerant (寬鬆比對) |
| 編號     | 配分       | 時間限制 | 敘述              |
| #0~#3  | 4%       | 0.1s | 所有食材對食材比例倍數相同   |
| #4~#99 | 96%      | 0.1s | 無特別限制           |

## B. 電梯巨星

終端雞大樓是個非常高聳的大樓，因為實在是太高了，因此大樓裡有一座電梯，能夠從1樓直達100000樓。有一天，巨星廚師雞在終端機電梯裡亂按按鈕，造成電梯不停上上下下，這時終端雞大樓的雞住戶都知道是廚師雞來了，許多雞都迫不及待地等在電梯前，祈禱電梯會停靠在自己所在的樓層。慶幸的是，電梯在上升或下降的過程中，經過的每個樓層都會停靠，因此大大增加了粉絲進入電梯的機會。

廚師雞為了提前準備足夠的布丁給粉絲們，於是透過偵查雞調查哪些樓層有雞在電梯門前等候，以此確定有多少雞能夠進入電梯拜訪牠。請你幫忙寫一個程式，計算有多少雞能夠進入電梯拜訪牠。

### 輸入格式

輸入共有三行，第一行有兩個整數 $n(1 \leq n \leq 100)$ ,  $m(1 \leq m \leq 100000)$ ， $n$ 代表電梯停靠的樓層數量， $m$ 代表雞的數量。第二行有 $n$ 個整數 $n_i(1 \leq n_i \leq 100000)$ ，代表電梯依序抵達的樓層。第三行有 $m$ 個整數 $m_i(1 \leq m_i \leq 100000)$ ，代表第 $i$ 隻雞所在的樓層。

### 輸出格式

輸出只有一行，包含一個整數 $n$ ，代表總共有 $n$ 隻雞可以進入電梯。

## 測試資料

|  |            |
|--|------------|
| 輸入範例1<br>2 5<br>1 10<br>2 5 9 13 17          | 輸出範例1<br>3 |
| 輸入範例2<br>2 5<br>10 1<br>2 5 9 13 17          | 輸出範例2<br>3 |
| 輸入範例3<br>5 5<br>10 1 15 12 16<br>2 5 9 13 17 | 輸出範例3<br>4 |
| 輸入範例4<br>5 5<br>6 7 5 6 4<br>1 8 3 10 15     | 輸出範例4<br>0 |

### 測試資料說明

範例測資1中，電梯會從1樓上升到10樓，所以在1~10樓的雞都能夠進入電梯，因此總共有3隻雞，分別是在2、5、9樓的雞。

範例測資2中，電梯從10樓下降到1樓，同樣是在1~10樓的雞能夠進入電梯，因此答案與範例測資1相同。

範例測資3中，電梯從10樓下降到1樓時，2、5、9樓的雞能夠進入電梯；電梯從1樓上升到15樓時，13樓的雞能夠進入電梯，因此總共有4隻雞能夠進入電梯，分別是2、5、9、13樓的雞。

範例測資4中，沒有雞可以進入電梯，因此輸出0。

## 配分

|         |          |      |                               |
|---------|----------|------|-------------------------------|
| 記憶體限制   | 64MBytes | 評分方式 | Tolerant (寬鬆比對)               |
| 編號      | 配分       | 時間限制 | 敘述                            |
| #0~#4   | 10%      | 1s   | $n=2$ 且 $m \leq 100$ ，且電梯只會上樓 |
| #5~#9   | 10%      | 1s   | $n=2$                         |
| #10~#29 | 80%      | 1s   | 無特別限制                         |

## C. 資料驗證

放山雞在進行資料傳輸或是提取資料時，為了確保資料的正確性，因此會進行資料驗證。其中一種資料驗證方式，是同位位元。同位位元驗證資料的方式分為兩種，奇核對位元以及偶核對位元。

以奇核對位元為例，給定一組資料中，若資料中1的個數為奇數，則補一個0，反之若資料中1的個數為偶數，則補一個1。以偶核對位元為例，給定一組資料中，若資料中1的個數為奇數，則補一個1，若資料中1的個數為偶數，則補一個0。

舉例來說，以1組7位元的資料組為例，對於「1000101」，補上奇核對位元為「10001010」，補上偶核對位元則為「10001011」，若驗證時的資料組，無法符合以上規則，我們就稱資料毀損。

今天給定 $n$ 組 $m$ 位元加1位同位位元資料，在 $k=1$ 時採用奇核對位元， $k=0$ 時採用偶核對位元，請對驗證正確的資料組輸出0，資料毀損的資料組輸出1，其中不含任何空白以及換行。

請依據以上規則，幫放山雞作出一套能驗證資料的程式。

### 輸入格式

第一行有三個正整數 $n, m, k$ ，其中 $n$ 為資料組組數且 $1 \leq n \leq 5000$ ， $m$ 為一組資料組的位元數(不包含同位位元)且 $1 \leq m \leq 10000$ ， $k$ 必為1或0， $k=1$ 時採用奇核對位元， $k=0$ 時採用偶核對位元，第二行有 $n \times (m+1)$ 個數字，而其中的數字必為0或1。

### 輸出格式

輸出只有一行，共有 $n$ 個數字，且每一個數字必為0或1，其中不包含任何空白以及換行。



## 測試資料

|   |                |
|---|----------------|
| 輸入範例1<br>1 7 1<br>10001010              | 輸出範例1<br>0     |
| 輸入範例2<br>1 7 0<br>10001010              | 輸出範例2<br>1     |
| 輸入範例3<br>5 3 1<br>10101011101010111010  | 輸出範例3<br>10101 |
| 輸入範例4<br>3 6 0<br>100101011011011111110 | 輸出範例4<br>110   |

## 測試資料說明

範例測資1、2請見題目敘述。

範例測資3中，「1010」的資料組為資料毀損的資料，輸出1，「1011」的資料組為驗證正確的資料，輸出0。

範例測資4中，「1001010」及「1101101」的資料組為資料毀損的資料，輸出1，「1111110」的資料組為驗證正確的資料，輸出0。

## 配分

|         |          |      |  |
|---------|----------|------|--|
| 記憶體限制   | 64MBytes | 評分方式 | Strictly (嚴格比對)                        |
| 編號      | 配分       | 時間限制 | 敘述                                     |
| #0~#1   | 6%       | 1s   | $n=1, 1 \leq m \leq 100$               |
| #2~#4   | 9%       | 1s   | $1 \leq n \leq 100, 1 \leq m \leq 100$ |
| #5~#9   | 33%      | 1s   | $1 \leq n \leq 1000$                   |
| #10~#19 | 52%      | 1s   | 無特別限制                                  |

## D. 市集採買

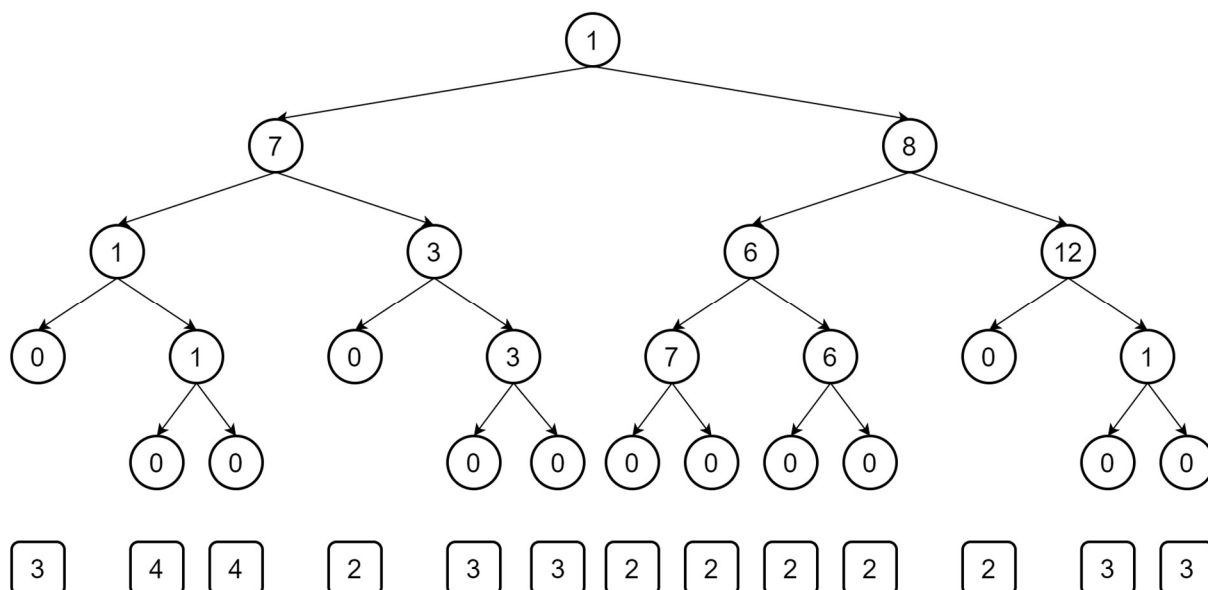
終端雞市集，是一個以二元樹為原型，打造出的市集，強大的設計理念，吸引許多攤販進駐。所有攤販都被安排在除了葉節點以外的節點，而葉節點只會有「已經到盡頭」的告示牌。攤販們被規定只能在攤位上賣一件物品，並且需要標明這件物品的大小，以便大家放心採買。

烏骨雞和放山雞放學後，決定去終端雞市集採買，烏骨雞想要盡可能在不超過背包容量的下，買到最多物品，但放山雞太懶惰，不想逛完整個市集，最後，兩雞達成協議，只會逛市集的其中一條路線，然後原路折返。

即便如此，烏骨雞還是想盡可能買到最多物品，因此牠想設計一個程式，輸入背包容量後，再以前序輸入市集攤販所賣物品之大小，就能獲得一條能買最多物品的路線。但因為可能同時存在多條路線，烏骨雞又有選擇困難，因此輸出永遠只會輸出最右邊那條。

以終端雞市集為例，假設烏骨雞背包容量為10，以前序輸入市集攤販所賣物品之大小為「17101003030086700600120100」，其中0為「已經到盡頭」的告示牌。而樹的圖如下圖所示，圓形內數字代表攤販所賣物品之大小，最底下的矩形內數字代表該條路在不超過背包容量下，所能買到最多物品數量，而輸出要選擇最多物品又最右邊的路線，所以輸出為「17110」。

由於烏骨雞能力不足，因此請你依所給條件，幫烏骨雞寫一個程式來達成目的。



## 輸入格式

第一行有一個不超過2147483647的正整數，代表烏骨雞的背包容量，第二行有 $2^n$ 個攤販所賣物品之大小，每個攤販所賣物品之大小為一個不超過100000的正整數。  
保證n不超過20且二元樹的深度不超過20層。

## 輸出格式

輸出只有一行，為一條能買最多物品且最右邊的路線。

## 測試資料

|   |                    |
|---|--------------------|
| 輸入範例1<br>7<br>3 4 0 0 5 0 0   | 輸出範例1<br>3 4 0     |
| 輸入範例2<br>10<br>1 7 1 0 1 0 0 3 0 3 0 0 8 6 7 0 0 6 0 0 1 2<br>0 1 0 0 | 輸出範例2<br>1 7 1 1 0 |

## 測試資料說明

範例測資1中，總共有四條路徑，由左到右分別為「3 4 0」「3 4 0」「3 5 0」「3 5 0」，前兩條路徑能買到最多物品數量為2，後兩條為1，又選擇最右邊的路徑，故輸出「3 4 0」。

範例測資2請見題目敘述以及圖片。

## 配分

|         |          |      |                 |
|---------|----------|------|-----------------|
| 記憶體限制   | 64MBytes | 評分方式 | Tolerant (寬鬆比對) |
| 編號      | 配分       | 時間限制 | 敘述              |
| #0~#10  | 11%      | 0.3s | 二元樹深度不超過5層      |
| #11~#29 | 19%      | 0.3s | 二元樹深度不超過10層     |
| #30~#52 | 23%      | 0.3s | 二元樹深度不超過15層     |
| #53~#99 | 47%      | 0.3s | 無特別限制           |

## F. 滿頭問號

烏骨雞使用通訊軟體與放山雞聊天時，總是會傳一些意義不明的訊息，讓放山雞滿頭問號。於是，放山雞決定在頭上的問號消失之前，將問號們打在聊天室傳給烏骨雞，藉此讓烏骨雞認知到自己的訊息有多麼令人不解。

放山雞的鍵盤上有四個功能鍵，分別是：

「

A 輸入一個問號

D 刪除一個問號

C 複製

V 貼上

」

其中C指的是複製當前所有的問號。

請你寫一個程式，幫助放山雞找出最少步驟打出特定問號數量的方法。

### 輸入格式

輸入只有一行，含有一個整數 $n$  ( $1 \leq n \leq 1000000$ )，代表放山雞要在聊天室中打出的問號數量。

### 輸出格式

請輸出使用最少步驟打出 $n$ 個問號的方法，請勿輸出空白或其他與題目無關之文字。**注意，答案可能不只有一種，輸出其中一種答案即可。**

## 測試資料

|              |                           |
|--------------|---------------------------|
| 輸入範例1<br>1   | 輸出範例1<br>A                |
| 輸入範例2<br>9   | 輸出範例2<br>AAACVV           |
| 輸入範例3<br>10  | 輸出範例3<br>AAACVVA          |
| 輸入範例4<br>179 | 輸出範例4<br>AAACVVVVCVVCVCVD |

## 測試資料說明

此說明僅說明範例輸出的結果。

範例測資1：題目要求打出1個問號，因此只需按下A鍵即可。

範例測資2：先用A鍵打出3個問號，再用C鍵複製，然後用V鍵貼上2次。

範例測資3：先以範例測資2的方法打出9個問號，再用A鍵補1個問號。

範例測資4：用A鍵打出3個問號，用C鍵複製，V鍵貼上4次，得到15個問號，C鍵複製，V鍵貼上2次，得到45個問號，C鍵複製，V鍵貼上1次，得到90個問號，C鍵複製，V鍵貼上1次，得到180個問號，最後用D鍵刪掉一個問號。

## 配分

|         |          |      |                |
|---------|----------|------|----------------|
| 記憶體限制   | 64MBytes | 評分方式 | Special (自訂比對) |
| 編號      | 配分       | 時間限制 | 敘述             |
| #0~#4   | 5%       | 1s   | $n \leq 10$    |
| #5~#19  | 15%      | 1s   | $n \leq 100$   |
| #20~#49 | 30%      | 1s   | $n \leq 10000$ |
| #50~#99 | 50%      | 1s   | 無特別限制          |