

# Problem 1. Sum of Odd Numbers 數字區間奇數和

(Time Limit: 2 seconds)

## 問題描述：

Given a range  $a$  to  $b$ , please find the sum of all the odd numbers within the range. For example, the sum of the odd numbers within the range  $[1,6]$  is  $1 + 3 + 5 = 9$ .

設定數字區間  $a$  到  $b$ ，請在  $a$ 、 $b$  區間內找到所有奇數之和。例如，區間的範圍為  $[1,6]$ ，則此區間的奇數和是  $1 + 3 + 5 = 9$ 。

## 輸入說明：

Given a number of datasets, where each dataset contains two numbers indicating the range, namely  $a$  and  $b$  ( $0 \leq a \leq b \leq 100$ ).

輸入若干組資料，每組資料均包含表達區間範圍的兩個數字， $a$  和  $b$  ( $0 \leq a \leq b \leq 100$ )。

## 輸出說明：

Output the sum of odd numbers within the range of each dataset.

請輸出每組資料其區間內所有奇數之和。

## 範例：

Sample Input:	Sample Output:
1 5	9
2 4	3
3 7	15
10 11	11
1 1	1

## Problem 2. Arithmetic Means of Two $N \times M$ Dimensional Arrays 兩個 $N \times M$ 維陣列的算術平均

(Time Limit: 3 seconds)

### 問題描述：

Write a program to take the arithmetic means of each pair of values in two  $N \times M$  dimensional arrays. These calculated means will be outputted into a newly generated  $N \times M$  dimensional array.

撰寫一個程式，計算兩個  $N \times M$  維陣列中的每對數值的算術平均，計算出的平均數將輸出到另一個新的  $N \times M$  維陣列中。

### 輸入說明：

The 1st line contains two integers,  $N M$  ( $1 \leq N, M \leq 3$ ). It represents that input data is  $N \times M$  array.

The following  $N \times M$  real numbers is array A.

The following  $N \times M$  real numbers is array B.

For example:

$N M$

$A[0, 0]$

$A[0, 1]$

$A[0, 2]$

...

$B[0, 0]$

$B[0, 1]$

$B[0, 2]$

...

第一行包含兩個整數， $N M$  ( $1 \leq N, M \leq 3$ )。它表示輸入數據是  $N \times M$  陣列。

接著的  $N \times M$  數字是陣列 A。

接著的 NxM 數字是陣列 B.

例如：

N M

A[0, 0]

A[0, 1]

A[0, 2]

...

B[0, 0]

B[0, 1]

B[0, 2]

...

#### 輸出說明:

A new NxM array whose elements are arithmetic means of array A and array B.

Each line represents a row, and output format for elements is [content].

There is no space between [content] and [content].

There is a blank line after each test case.

※ round to one decimal places

一個新的 NxM 陣列，其元素是陣列 A 和陣列 B 的算術平均數。

每行輸出代表陣列的一列，元素的輸出格式為[content]。

[content]和[content]之間沒有空格。

每個測試用例後都有一個空行。

※ 舍入到小數點後一位

範例:

Sample Input:	Sample Output:
2 2	[5.0][5.0]
2 3	[4.0][5.0]
7 1	
8 7	
1 9	
3 2	[4.0][5.0]
6.5 3	[5.0][5.0]
1 2	[5.0][4.0]
4 1	
1.5 7	
9 8	
6 7	

## Problem 3. Fake Coins!! 偽造的金幣!!

(Time Limit: 2 seconds)

### 問題描述：

Among  $n$  bags of gold coins, one of them is counterfeit. We do not know which one is counterfeit, but we know counterfeit gold coins are lighter than real gold coins. Now we have a scale without scales that you can put more than one bags of gold coins on it. You need to weigh for a few times, then you can find out which bag of gold coins is counterfeit.

在  $n$  袋金幣中，有一袋金幣是偽造的，但我們不知道哪一袋金幣是偽造的金幣，我們知道偽造的金幣比真的金幣還輕，而手邊又正好只有沒有刻度的天秤，天秤的一端可放多袋金幣，聰明的你最多需要使用天秤幾次，保證一定能找出哪一袋是偽造的金幣。

### 輸入說明：

The first line of the input contains an integer  $N(0 < N \leq 20)$  indicating the number of test cases. For each test case, there is a line with one integer numbers  $n(2 \leq n \leq 1000000)$  representing how many bags of gold coins.

第一行為一個整數  $N(0 < N \leq 20)$  代表有  $N$  組測試資料，之後會有  $N$  行數字，每一行數字代表  $n(2 \leq n \leq 1000000)$  袋金幣。

### 輸出說明：

For each test case, the output is the maximum number of times you use the scale. Add a "newline" in the end of the output.

輸出  $N$  組測試資料可使用天秤的最多次數，最後必須有換行字元。

### 範例：

Sample Input:	Sample Output:
3	2
8	3
17	5
100	

## Problem 4. Decimal to Binary 十進制轉二進制

(Time Limit: 2 seconds)

### 問題描述：

Write a program which converts each input decimal number into the corresponding 8-bit binary number. If the input number is negative, please use two's complement notation.

撰寫一個程式，使用者輸入一個整數，印出 8 位元的二進制表示。如果輸入為負數，則以二補數表示。

### 輸入說明：

In the first line, input a positive integer number  $n$  ( $n \leq 50$ ), and in the following lines, input  $n$  integer numbers with each range between -128~127.

首先輸入一個正整數  $n$  ( $n \leq 50$ )，表示接下來有  $n$  筆測資。每筆測資輸入數個整數，介於 -128 ~ 127 之間。

### 輸出說明：

Show the corresponding binary number in length of 8 bits. Please print a carriage return at the end of each output line.

以 8 位元的二進制顯示，最後必須有換行字元。

### 範例：

Sample Input:	Sample Output:
2	00001111
15	11111110
-2	

## Problem 5. Covering a Hole 覆蓋孔洞

(Time Limit: 3 seconds)

### 問題描述：

Tom works in a company that produces covers for all kinds of holes, such as holes on streets and wells. He encounters a problem as follows: given a hole  $H$  which is a polygon with interior angles of only 90 or 270 degrees, determine the smallest rectangular cover that can completely cover  $H$ . In this problem,  $H$  is given in a coordinate system such that each of its edges is either vertical or horizontal. When covering a hole, each edge of the cover should also be either vertical or horizontal in the same coordinate system.

Consider the example in Figure 1. It is easy to see that the smallest rectangular cover that can completely cover  $H$  is a rectangle of size  $4 \times 8$ .

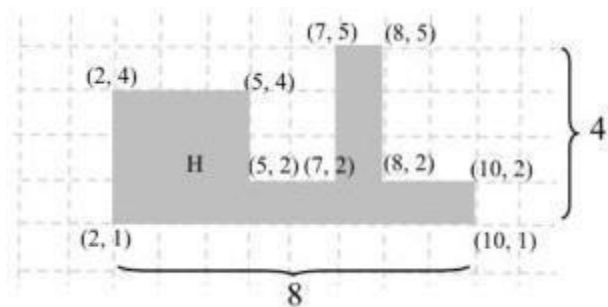


Figure 1: a rectangular hole  $H$ .

In this problem, you are asked to find the area of the smallest rectangular cover that can completely cover  $H$ . For example, in Figure 1, the output is 32.

### Technical Specification

1. The number of the vertices of  $H$ , denoted by  $n$ , is a positive integer between 4 and 100.
2. The x-coordinates and y-coordinates of vertices are integers between 0 and 1000.

湯姆在一家公司工作，該公司生產各種孔洞的蓋子，例如街道和水井的孔洞。他遇到以下問題：給定一個孔洞  $H$ （該孔洞是一個僅具有 90 度或 270 度內角的多邊形），請找出可以完全蓋住  $H$  的最小矩形蓋子。在此問題中， $H$  的坐標是由每個邊緣均為垂直或水平的坐標系統表示。覆蓋孔洞時，蓋子的每個邊緣在同一坐標系統中也應為垂直或水平。

考慮圖 1 中的示例。很容易看出，可以完全覆蓋  $H$  的最小矩形蓋子是大小為  $4 \times 8$  的矩形。

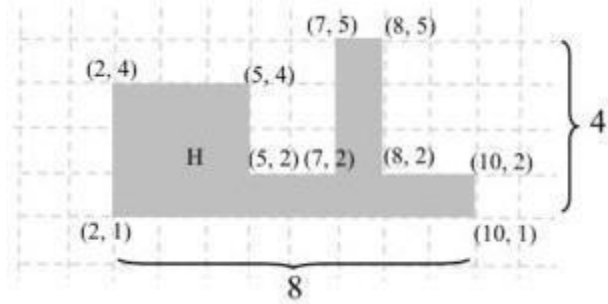


圖 1：矩型孔洞 H

在此問題中，你被要求找出可以完全覆蓋 H 的最小矩形蓋子。例如，在圖 1 中，輸出為 32。

特別定義：

1. H 的頂點數 (用  $n$  表示) 是 4 到 100 之間的正整數。
2. 頂點的  $x$  坐標和  $y$  坐標是 0 到 1000 之間的整數。

### 輸入說明:

The first line is an integer  $t$ ,  $1 \leq t \leq 10$ , indicating the number of test cases.

Each test case starts with one line containing the number  $n$ ,  $4 \leq n \leq 100$ , of vertices of the hole H. Then,  $n$  lines follow, each of which includes two integers  $x$  and  $y$ ,  $0 \leq x, y \leq 1000$ , which are the coordinates of the vertices of the hole's polygon. In the order, they would be visited on a trip around the polygon.

第一行是整數  $t$ ，其中  $1 \leq t \leq 10$ ，表示測試案例的數量。每個測試案例開始的第一行為整數  $n$  ( $4 \leq n \leq 100$ )，代表孔洞 H 的頂點數量。接著是  $n$  行數值，每行包含兩個整數  $x$  和  $y$ ，其中  $0 \leq x, y \leq 1000$ ，表示孔洞多邊形的頂點坐標，依照在多邊形周圍繞一圈的順序讀取它們。

### 輸出說明:

For each test case, output the area of the smallest rectangular cover that can completely cover H in one line.

對於每個測試案例，輸出一行可以完全覆蓋 H 的最小矩形蓋子的面積。



範例:

Sample Input:	Sample Output:
2	32
10	9
10 1	
10 2	
8 2	
8 5	
7 5	
7 2	
5 2	
5 4	
2 4	
2 1	
4	
2 1	
5 1	
5 4	
2 4	

## Problem 6. Recursive Program Exercise

### 遞迴程式練習

#### 問題描述：

Given the following recursive function:

$$f(n) = \begin{cases} n+1, & \text{when } n = 0, n = 1 \\ f(n-1) + f\left(\left\lfloor \frac{n}{2} \right\rfloor\right), & \text{when } n > 1 \end{cases}$$

The floor function is the function that takes as input a real number  $x$ , and gives as output the greatest integer less than or equal to  $x$  denoted  $\text{floor}(x)$  or  $\lfloor x \rfloor$

Floor example :

x	Floor
2	2
2.4	2
2.9	2
-2.7	-3
-2	-2

給定下列遞迴函式：

$$f(n) = \begin{cases} n+1, & \text{when } n = 0, n = 1 \\ f(n-1) + f\left(\left\lfloor \frac{n}{2} \right\rfloor\right), & \text{when } n > 1 \end{cases}$$

取整函數是一類將實數對映到相近的整數的函數。常用的取整函數有兩個，分別是下取整函數和上取整函數。下取整函數即為取底符號，在數學中一般記作  $\lfloor \cdot \rfloor$  或在電腦科學中一般記作  $\text{floor}(x)$ ，表示不超過  $x$  的整數中最大的一個。

Floor 範例：

x	Floor
2	2
2.4	2
2.9	2
-2.7	-3
-2	-2

Please calculate  $f(k)$ .

請計算出  $f(k)$ 。

**輸入說明:**

The input value is an integer greater than 1.

輸入值為一個大於 1 的整數。

**輸出說明:**

The calculation result of  $f(k)$ .

輸出  $f(k)$  的計算結果。

**範例:**

Sample Input	Sample Output
10	60
12	94