**6 тур - з 17.12 по 24.12.2018**

точка входу для відправлення розв'язків
[http://134.249.159.199//cgi-bin/new-client?contest\_id=67](http://134.249.159.199/cgi-bin/new-client?contest_id=67)

**Задача A . (100 балів)**

**Ім’я вхідного файлу: input.txt**

**Ім’я вихідного файлу: output.txt**

**Обмеження по часу: 1 с**

**Обмеження по пам’яті: 64 megabytes**

Цього року в Івана Семеновича яблука вродили на славу. Він зібрав усі яблука у велику корзину та віддав своїм ***К*** внукам.

Перший внук взяв з корзини половину всіх яблук і ще **а1** яблуко (*якщо кількість яблук не ділилась на 2 націло, то результат ділення на 2 він міг заокруглити як у більшу, так і в меншу сторону*). Для прикладу, якщо в корзині було **7** яблук і **а1=1**, то він міг взяти або **4**, або **5** яблук; якщо ж було **6** яблук і **а1=1**, то він міг взяти лише **4** яблука.

Другий внук взяв половину від того, що залишилося, і ще **а2** яблук (*якщо кількість яблук була непарна, то половину він міг заокруглити як у більшу, так і у меншу сторону*). І так далі, ***К-ий*** внук взяв половину від того, що залишилося після ***К-1*** внука, і ще **ак** яблук. В підсумку корзина виявилася порожньою.

Після цього вони задумалися, наскільки ж великий був урожай яблук у дідуся. Жоден з них не пам’ятав, чи ділилась націло на **2** кількість яблук в його випадку, а якщо ні, то в яку сторону було зроблено заокруглення.

Отож, внуків цікавить мінімальна та максимальна кількість яблук, яка могла б бути в корзині на початку.

 **Формат вхідних даних**

Спочатку задано ціле додатне число ***К*** ($1\leq K\leq 1000$). Далі записано ***К*** цілих невід’ємних чисел ***а1, а2, …, ак***. ($0\leq a\_{k}\leq 1000$).

**Формат вихідних даних**

Виведіть два цілих невід’ємних числа, кожне в новому рядку – мінімальну та максимальну кількість яблук в корзині відповідно.

**Наприклад,**

|  |  |
| --- | --- |
| **input.txt** | **output.txt** |
| 20 1 | 17 |
| 11 | 13 |

**Задача A . (100 балів)**

**Ім’я вхідного файлу: input.txt**

**Ім’я вихідного файлу: output.txt**

**Обмеження по часу: 1 секунда**

**Обмеження по пам’яті: 64 megabytes**

Після того, як закінчилися усі осінні роботи, Іван Семенович вирішив замовити у свій заміський будинок новий LCD-телевізор. Фірма, що виконувала замовлення, привезла телевізор у дуже великій кубічній коробці розміром 1х1х1 метр. Оскільки автомобіль не зміг під’їхати до будинку, вантаж було розвантажено біля воріт. Одна зі сторін коробки (*мається на увазі грань куба*) була позначена як крихка – та, поряд з якою був розташований екран. Вантаж було розвантажено так, щоб крихка сторона не знаходилася на землі.

Через великі розміри коробку по подвір’ї можна було лише перекочувати через ребра (*тобто, класти на грань куба*). При цьому крихка сторона не повинна була знаходитись на землі, інакше можна було пошкодити екран.

Подвір’я має форму прямокутника розміром ***N*** на ***M*** метрів. План вулиці нарисований на папері в клітинку, причому розмір клітинки рівний 1 метрові. На плані введена система координат так, що ліва нижня клітинка схеми має координату ***(1;1)***, права нижня – ***(1;M)***, права верхня – ***(N,M)***.

Коробка була розвантажена поряд з ворітьми, в клітинці, що має координати ***(1;b)*** (*ця клітинка розміщена в нижній частині на плані ділянки*)*,* а перекотити її необхідно до дверей будинку – клітинку, що має координати***(c;d)***. На початках вказано, з якої сторони знаходиться крихка сторона, а з якої сторони вона буде після перекочування, не важливо, – основне, щоб вона не опинилася на землі. Ділянка по периметру огороджена парканом, тому викотити її за межі ділянки не вдасться.

Допоможіть вантажникам перекотити коробку від воріт до дверей будинку.



**Рис. 1 для першого прикладу**

**Формат вхідних даних**

У першому рядку знаходяться цілі числа **N, M, b, c, d** (***1≤N≤10000, 1≤M≤10000, 1≤b≤M, 1≤c≤N, 1≤d≤M***). У другому рядку міститься одна з літер **L**, **R**, **T**, **F**, **B**, що визначають початкове положення крихкої сторони коробки (зліва, справа, зверху, спереду, ззаду відповідно). Вважається, що задня сторона пакунку повернута в бік воріт. Ворота та двері на плані позначені різними клітинками.

**Формат вхідних даних**

Виведіть послідовність перекочувань, яка дозволить вантажникам виконати поставлене завдання. Перекочування позначається літерами:

**L** – перекочування вліво (на одиницю зменшується друга координати);

**R** – перекочування вправо (на одиницю збільшується друга координати);

**F** – перекочування вперед (на одиницю збільшується перша координати);

**B** – перекочування назад (на одиницю зменшується перша координати);

Загальна кількість перекочувань не повинна перевищувати ***4(M+N)*** – в іншому випадку вантажники відмовляться від даної роботи.

Якщо завдання виконати неможливо – виведіть **IMPOSSIBLE**.

**Наприклад,**

|  |  |
| --- | --- |
| **input.txt** | **output.txt** |
| 4 3 2 3 2T | RFFL |
| 2 1 1 2 1F | IMPOSSIBLE |