P1: Reverse and Add *

Problem

把一個數字反轉並相加的方法很簡單：就是把數字反轉並加上原來的數字。假如這個和不是一個迴文（指這個數字從左到右和從右到左都相同），就一直重複這個程序。假設開始的數字是 195，轉換過程如下：

195	786	1473	5214
591	687	3741	4125
----- →	----- →	----- →	-----
786	1473	5214	9339

9339 迴文數字出現了，在這個例子中，經過了 4 次相加後得到了迴文 9339。幾乎對所有的整數這個方法都會得到迴文，但是也有有趣的例外。196 是第 1 個用這個方法找不到迴文的數字，然而並沒有證明該迴文不存在。現在給你一個開始的數字，你的任務就是求出經過多少次相加後，會產生哪一個迴文。對所有的測試資料，你可以假設：

1. 都會有 1 個答案。
2. 在 1000 個相加內都會得到答案。
3. 產生的迴文數字 <= 2^{64}-1.

Input

第 1 列有一個整數 N（0 < N <= 100），代表以下有幾組測試資料。每筆測試資料一列，各有 1 個整數 P，就是開始的數字。

Output

對每一測試資料，請輸出 2 個數字：得到迴文所需的最少次數的相加，以及該迴文。

Sample Input　　Sample Output

4↵ 195↵ 265↵ 750↵ 9229↵	4 9339↵ 5 45254↵ 3 6666↵ 0 9229↵

P2: Science - day of the week *

Problem

Zeller's congruence is an algorithm developed by Christian Zeller to calculate the day of the week. The formula is

$$h = \left(q + \frac{26(m+1)}{10} + k + \frac{k}{4} + \frac{j}{4} + 5j\right)\%7$$

,

where

- ***h*** is the day of the week (0: Saturday, 1: Sunday, 2: Monday, 3: Tuesday, 4: Wednesday, 5: Thursday, and 6: Friday).
- ***q*** is the *day* of the month.
- ***m*** is the *month* (3: March, 4: April, . . ., 12: December). January and February are counted as months 13 and 14 of the previous year. If *month* is 1 (or 2), replace it with 13 (or 14) and decrease *year* by 1.
- ***j*** is *year*/100.
- ***k*** is the year of the century (i.e., *year* % 100).

Note all divisions in this problem perform an integer division. Write a program that prompts the user to enter a year, month, and day of the month, and

Input

Each line inputs 3 numbers denoting *year*, *month*, and day of the month, respectively. There are many lines of inputs.

Output

For each test case, print out the name of the day of the week.

Sample Input Sample Output

2015 1 25↲	Day of the week is Sunday.↲
2012 5 12↲	Day of the week is Saturday.↲
2021 10 11↲	Day of the week is Monday.↲
2021 10 26↲	Day of the week is Tuesday.↲

P3: BigDecimal Calculator **

Problem

Write a program to input an expression string in which the operands and operators are separated by zero or more spaces. For example, 3.5*4+3 and 3.5 + 4 % 3 are acceptable expressions. The operator in the expression might be +, -, *, /, and %. Your program must print out the expression and its computing result. The sample output for the input expression 3.5*4+3 is shown below:

3.5 * 4 + 3 = 17

Requirement

Write a static method BigDecimal calculate(String exp) to compute the expression and return a BigDecimal result. The operands should be stored as BigDecimal in this method. You have to use the arithmetic operators provided by the BigDecimal class to calculate the expression. **(未依規定，以 0 分計)**

Input

There are many input lines. Each line has an input expression *Exp*. There are three operands and two operators in *Exp*.

Output

For each input expression *Exp*, please output the expression and its computing result. Note that all tokens are separated by a space character.（小數點以下印一位）

Sample Input　　Sample Output

3 + 4↲	3 + 4 = 7.0↲
32.5-20.5*2↲	32.5 - 20.5 * 2 = -8.5↲
4 * 5.6 + 1.1↲	4 * 5.6 + 1.1 = 23.5↲
20.4 / 4 -3.1↲	20.4 / 4 – 3.1 = 2.0↲
20.8 % 4.1 + 1.8↲	20.8 % 4.1 + 1.8 = 2.1↲
-21.5↲	-21.5↲
0.0+0.0↲	0.0↲

P4: Magic Square **

Problem

If you have good observations skills, you may found that building a Magic Square is simple. A Magic Square has only an odd number N of rows and columns where N < 100. A Magic Square is created by integer numbers in the range from 1 to N^2 , with a peculiar property, the “sum of the numbers” in each row, column and diagonal are the same.

For the case n = 3,

M. Square	Rows	Columns	Diagonals
4 9 2	4+9+2 = 15	4+3+8 = 15	2+5+8 = 15
3 5 7	3+5+7 = 15	9+5+1 = 15	4+5+6 = 15
8 1 6	8+1+6 = 15	2+7+6 = 15	

Input

Each line contains an Integer N denoting an N * N Magic Square.

Output

如果 N 是偶數則輸出 ”It is not an odd number. ”。

如果 N 是奇數則先輸出直橫列的加總數字，再輸出 N * N 數字陣列,每個數字以%5d 格式輸出。每組測資間請空一行。

Sample Input　　Sample Output

4↵ 3↵	It is not an odd number.↵ ↵ 15 4 9 2↵ 3 5 7↵ 8 1 6↵

P5: Prime Factorization ***

Problem

輸入數字 *N* (資料型態為 Integer)， 請輸出該數字的所有質因數及其次方。例如 $N=360=2^3*3^2*5$。此題數字可能會有質數出現。

Requirement

請撰寫以下兩個 static methods： **(未依規定，以 0 分計)**

1. **boolean [] PrimeArray(long N) {….}**
   which returns an array *A* of Boolean values, where *A*[i] is true if i is a prime number, otherwise, *A*[i] is false if i is not a prime number. Note that *A*.length = N+1;
   Hint: if n is a prime number, then n * j is not a prime, where j >= 2;
2. **String PrimeFactorization(long N) {…}**
   which returns a string of prime factorization for the number N. For example, if N = 360, the returned string is "2^3 * 3^2 * 5".

Input

輸入有多列，每列有個整數 N，最多 1000 列。

Output

第一行輸出所有數字中之最大數 X 及其開根號整數$\sqrt{X}$，其後針對每一組測資數字 N，輸出 N 的質因數分解，將數字 N 的所有質因數（及其次方）以小到大方式顯示出來，如質因數之次方數大於 1，以^運算符號顯示，不同質因數間以 * 運算符號互相連接， *運算符號前後加空格。

Sample Input　　Sample Output

360↵ 3072↵ 23↵	3072 55 2^3 * 3^2 * 5↵ 2^10 * 3↵ 23↵