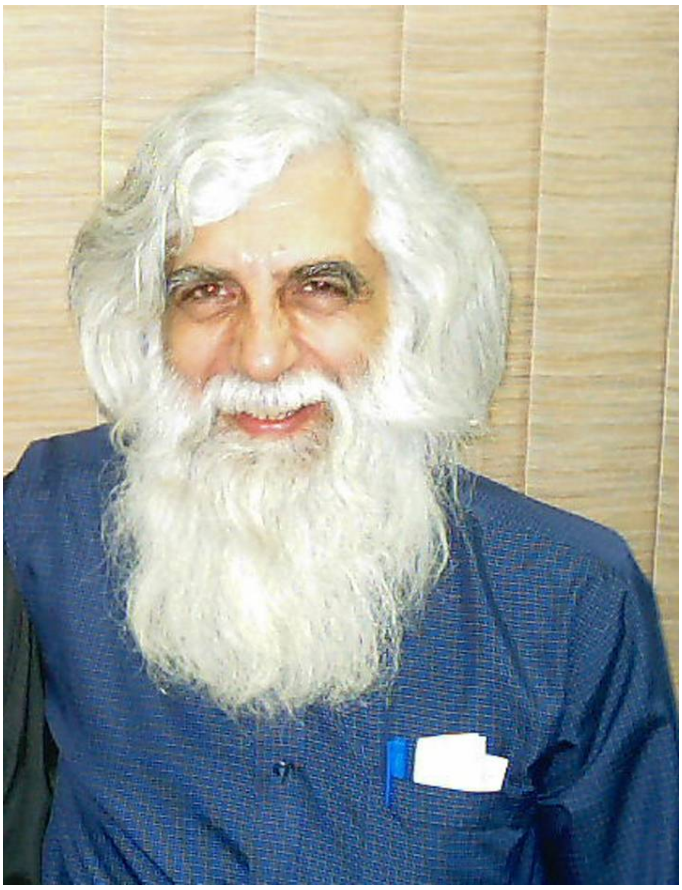


به یاد استاد فقید

پروفسور کارو لوکاس

(۱۳ شهریور ۱۳۲۸— ۱۷ تیر ۱۳۸۹)

پدر علم رباتیک ایران و چهره ماندگار مهندسی کشور در سال ۱۳۸۵

"هدف از کلاس، انتقال اطلاعات نیست، بلکه انجام تحقیقی مشترک است."

A.

جومونگ پس از تاسیس امپراطوری گوگوریو، برای این کشور الفبای جدیدی را ابداع کرده که شامل ۱۰ حرف است. حروف این الفبا به ترتیب از چپ به راست بصورت زیر است:

+) s 7 { p 5 8 ! k

می خواهیم برنامه ای بنویسیم که کلماتی به زبان گوگوریویی از ورودی بگیرد و حروف این کلمه را مرتب کرده و در خروجی چاپ کند. به عنوان مثال اگر کلمه ورودی بصورت p{+! باشد خروجی مرتب شده بصورت +{p! خواهد بود. اگر در کلمه ورودی حرفی وجود داشته باشد که جزو الفبای گوگوریویی نباشد در خروجی کلمه what???!! چاپ شود.

نکته: هر کلمه گوگوریویی حداکثر ۱۵ حرف دارد.

نمونه ورودی	نمونه خروجی
p{+!	**+{p!**
5k)s!7	**)s75!k**
s5ke7	**What???!!**

B.

محمد علیمیرزالو دبیر انجمن علمی علوم کامپیوتر بی صبرانه منتظر تابستان است تا در تعطیلات تابستان در کلاس های ورزشی و آموزشی شرکت کند. او می خواهد در تعدادی از کلاس ها ثبت نام کند اما مشکل او این است که برخی از کلاس های مورد نظر او ممکن است از نظر زمانی با یکدیگر تداخل داشته باشند. شما باید به او کمک کنید تا ببیند که آیا کلاس های مد نظر او تداخل زمانی دارند یا خیر.

ورودی برنامه:

حداکثر تعداد کلاس هایی که باید بررسی شوند ۵ کلاس است (بین ۲ و ۵). در خط اول ورودی شما تعداد کلاس هایی که ساعات آنها باید بررسی شوند وارد می کنید. به عنوان مثال با ورود عدد ۳، تداخل ساعات آغاز و پایان ۳ کلاس باید بررسی شوند. در ادامه ساعات آغاز و پایان هر کلاس در سطرهای مجزا وارد می شوند. زمان های وارد شده از دو قسمت ساعت(دو رقم بین 00 تا 23) و دقیقه (دو رقم بین 00 تا 59) تشکیل شده اند که با **:** از هم جدا شده اند. بین ساعت شروع و پایان یک خط تیره (-) درج می شود.

خروجی برنامه:

خروجی برنامه عبارت "No Conflict" برای عدم تداخل ساعات و " Conflict" برای تداخل ساعات می باشد.

نمونه ورودی	نمونه خروجی
2 **09:00-10:30** **11:30-13:25**	**No Conflict**
3 **14:20-16:10** **08:15-10:25** **15:30-17:00**	**Conflict**

هر سال شمسی غیر کبیسه از ۶ ماه ۳۱ روزه، ۵ ماه ۳۰ روزه و ۱ ماه ۲۹ روزه تشکیل شده است. می خواهیم برنامه ای داشته باشیم که با وارد کردن تاریخ بصورت ماه و روز، مشخص کند که آن روز چه روزی از هفته است. فرض کنید روز اول سال (۱/۱) روز یکشنبه است.

ورودی:

دو عدد در ورودی خواهیم داشت که عدد اول نشان دهنده ماه و عدد دوم نشان دهنده روز است.

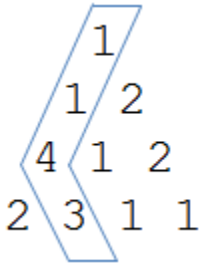
خروجی:

یکی از روزهای هفته (Sunday – Monday – Tuesday – Wednesday – Thursday – Friday – Saturday)

نمونه ورودی	نمونه خروجی
۱ ۲	**Monday**
۲ ۳۰	**Thursday**
۱۰ ۱۸	**Saturday**

D

You are given a triangle of positive integers, such as appears on below.



The triangle is composed of *N* rows. The first row contains a single number; the second row contains two numbers; the third row contains three numbers, and so on. Define a *path* in this triangle as starting from the number on the first row and ending at a number on the N^{th} row. The path extends downward by either moving to the number immediately below and to the left, or below and to the right. In the example highlighted on the right, the path starts at 1 on the first row and extends to the 3 on the N^{th} row. The sum of the numbers in this path is 9; no other path in the above triangle has a greater sum, thus 9 is the maximal sum in this triangle. Given a triangle of numbers with *N* rows, write a program to compute the maximal sum of any such path. Note that each path contains *N* numbers.

Input

The first line of input is a positive integer *N* (1 < n <= 8) on a line by itself representing the number of rows in the number triangle. The next *N* lines contain the positive integers of each successive row of the triangle, with a space between each integer. You can assume that the triangle is properly encoded in the input.

Output

Output consists of a single positive integer on a line by itself representing the greatest sum of all such paths in the triangle.

Sample Input and Output

input	output
3 1 2 1 1 2 3	5
4 1 1 2 4 1 2 2 3 1 1	9
6 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1 1 5 10 10 5 1	23
6 1 1 1 1 1 2 3 3 1 1 1 1 4 4 6 10 10 5 5 1	17

E.

Your task is to write a program which determines if word pairs given in an input are anagrams of one another. Anagrams are words which contain exactly the same letters in the same frequency, but in a different order. Example: the word read and the word dare are anagrams.
The input data will contain one line of data for each pair of words to be analyzed, with a single blank between the two words. For example, if the input contained these word pairs, it might look like:

read dare
stake takes
tofu tofu
pumpkin pear
one two

The first two word-pairs are anagrams, the last three pairs are not. You may assume that: the input contains at least one word-pair, that each line contains two valid English words consisting of only lower case letters and that a word will contain no more than 20 characters. You may assume all data provided is valid and that you do not have to do any bad data checking on this problem.
Your output for the example output would look like:

YES!
YES!
no!
no!
no!

Input	Output
read dare	YES!
stake takes	YES!
tofu tofu	no!
pumpkin pear	no!
one two	no!

F

A numerologist might compute a number from a word as follows: assign the letter **A** a value of 1, the letter **B** a value of 2, …, and the letter **Z** has a value of 26. Now sum up the values of each letter in a word. For example, the word **CAB** has three letters and its numerological total is 3+1+2 = 6.
However, instead of computing the sum, what if you were asked to compute the *Alphabetic Average*. That is, compute the sum as shown above, and then divide by the total number of letters in the word. Since this average might not be a proper integer, you need to round it to the nearest integer, whose value (as before) determines a letter, which is the *alphabetic average* of the original word. When rounding use the following rule:

- To round to the nearest integer a positive number *x* whose fractional value is 0.5 or higher, choose the smallest integer greater than *x*, otherwise choose the largest integer smaller than *x*

For example, 1.4999 rounds (down) to the integer 1, while 13.5 rounds (up) to 14.
For **CAB**, the average (3+1+2)/3 = 2 which corresponds to **B** being the alphabetic average.
Here is an example using the word **AWAY**:
(1 + 23 + 1 + 25)/4 = 50 / 4 = 12.5 and this rounds to 13, which corresponds to the letter **M**

Input

The input contains a single word on a line by itself composed only from capital letters A to Z. The word will contain no more than 20 letters.

Output

The output consists of a single capital letter on a line by itself.

Sample Input and Output

Input	Output
CAB	B
SCARE	I
AWAY	M
AN	H

. G

A *palindrome* is a sequence of letters that reads the same forwards and backwards. You are given a word phrase containing just capital letters and spaces. If you consider just the vowels in the word – A, E, I, O, U, or Y – do they form a palindromic sequence?
For example, the phrase "WHAT IS THAT" contains three vowels, which when read from left to right in the phrase are "AIA"; this forms a vowel palindrome.
Here's another example, "ONE TO HIT ON ELBOW" contains the vowels (in this order) "OEOIOEO" which is a vowel palindrome.

Input

The input consists of a single line of characters containing capital letters and spaces. There will be no more than 40 characters in the input.

Output

The output consists of the phrase "X IS PALINDROME" or "X IS NOT A PALINDROME" where "X" represents the vowels in the original phrase. All letters in the output must be capital letters.

Sample Input and Output

Input	Output
WHAT IS THAT	"AIA" IS PALINDROME
ONE TO HIT ON ELBOW	"OEOIOEO" IS PALINDROME
WHY NOT UNDO IT	"YOUOI" IS NOT A PALINDROME

H

A Numerologist would like you to write a program to detect **special years** and he has provided the following computation. For a year with four digits, create a mathematical integer computation using just the basic operators – addition (+), subtraction (–), multiplication (*) and division (/) – between the first three digits and compute whether the result is equal to the fourth digit, which would make the year special. For example:

year	Is special?	Reason
1991	YES !	1 + 9 – 9 = 1
1891	YES !	1 + 8 / 9 = 1
2484	NO !	no arrangement of operators and digits works
1482	NO !	no arrangement of operators and digits works

In all cases, the computation proceeds exactly as it would use a hand-held calculator. Thus the year 1891 is special because 1+8 equals 9 which can be divided by 9 to equal 1. However, the year 2484 is ordinary because 2/4 is not an integer and no other arrangement of operations can be used to declare the year special.
Your task is to write a program that determines whether a year is SPECIAL or ORDINARY.

Input

The input consists of a single line with an integer N by itself, representing a year between 1000 and 9999 inclusively.

Output

The output will consist of a single line containing the word “SPECIAL” or “ORDINARY” by itself.

Sample Input and Output

Input	Output
1991	SPECIAL
1891	SPECIAL
2484	ORDINARY
1482	ORDINARY

.I

Alice recently read the news about Edward Snowden, and learned that encryption could be used to protect personal communication effectively. She decided to construct a simple encryption scheme to protect her own communications as follows. First, she generates a set of randomly chosen "secret keys" K1, K2, K3. Next, using these keys, she encrypts a set of messages M1, M2, M3, obtaining ciphertexts C1= K1 + M1, C2= K2+ M2, C3 =K3+M3.
In implementing her scheme, she decides to choose the secret keys from the set A={1, 2, …, 100}, and the messages from the sequence B=(a, b, c, …, z). She is adamant that her ciphertexts should *also* be elements of the sequence B; to this end, she defines the ``+'' operator in expressions such as C1=K1+M1 to *shift* element M1 in B to the right by K1 positions. For example, if K1=1 and M1= b, then the corresponding ciphertext is C1= c. Similarly, if K1=2 and M1=b, then the ciphertext is C1=d, if K1=24 and M1=b, the ciphertext is C1=z, and if K1=25 and M1=b, then the ciphertext is C1=a.

Input

The input consists of four rows.

1. The first row is an integer n satisfying 1 <= n <= 100 specifying the number of secret keys.
2. The second row is a sequence of n integers, each from the set A={1,2,…,100}, corresponding to the secret keys.
3. The third row is a sequence of symbols from the set B={a,b,…,z} corresponding to the messaged to be encrypted.
4. The fourth row is a special value -1 to indicate the end of the input phase.

Output

Output a single row of symbols from set B corresponding to each of the ciphertexts.

Sample Input and Output

Input	Output
3 1 2 3 bzy -1	cbb
5 3 4 4 1 2 hello -1	kipmq