Intra University ICCP Junior Programming Contest

A. Hello ICCP!

Output file: Standard Output
Time limit: 1 Seconds
Memory limit: 512 Megabytes

Sourav wants to join the competitive programming community of **Northern University of Business and Technology Khulna**, known as the **"Interested Community for Competitive Programming" (ICCP)**. However, there is a requirement to enter the community: he must write a program that outputs "**Hello, ICCP!**". As he is new to programming, let's help him write the program.

Output

Just output **"Hello, ICCP!"** without double quotation marks.

Standard Output
Hello, ICCP!

B. House of the Dragon

Input file: Standard Input
Output file: Standard Output
Time limit: 1 Seconds
Memory limit: 512 Megabytes

House Targaryen of King's Landing is a noble family of ***Valyrian*** descent who once ruled the ***Seven Kingdoms*** of ***Westeros***. The Targaryen sigil is a red three-headed dragon, breathing red flames on black.The house words are ***"Fire and Blood"***.
The Dance of the Dragons was a civil war during ***Targaryen*** rule of the ***Seven Kingdoms***. A war of succession between ***Aegon II known as "The Greens"*** and his half-sister ***Rhaenyra known as "the blacks"*** over their father ***Viserys I's*** throne, the war was fought from 129 AC to 131 AC.

No one really knows how many dragons that participated in ***the Dance of the Dragons***. Different people say different numbers. In the civil war, all fully grown dragons participated in ***the Dance of the Dragons***. Some people say ***the Greens*** had **x** dragons and the Blacks had **y** dragons.
As a great fan of ***House of the Dragon***, ***Arko*** wants to know the total number of dragons that participated in ***the Dance of the Dragons***. Write a program to help him figure out the **total number of dragons** that participated in ***the Dance of the Dragons***.

Input

The single line of each test case contains two non-negative integer $\mathbf{x}$ ($1 \leq \mathbf{x} \leq 10$) and $\mathbf{y}$ ($1 \leq \mathbf{y} \leq 10$).

Output

Output the total number of dragons that participated in **the Dance of the Dragons**.

Standard Input	Standard Output
1 2	3

Standard Input	Standard Output
5 3	8

Standard Input	Standard Output
2 2	4

Explanation

In the first table:
The Greens has 1 dragon and The Black has 2 dragons.
The total number of dragon = 1 + 2 = 3

In the second table:
The Greens has 5 dragons and The Black has 3 dragons.
The total number of dragon = 5 + 3 = 8

In the third table:
The Greens has 2 dragons and The Black has 2 dragons.
The total number of dragon = 2 + 2 = 4

C. Carl Friedrich Gauss Intelligence

Input file: Standard Input
Output file: Standard Output
Time limit: 1 Seconds
Memory limit: 512 Megabytes

Carl Friedrich Gauss is commonly known as the ***"Prince of Mathematicians"*** due to his profound and far-reaching contributions to the field of mathematics. This title reflects the depth, breadth, and lasting impact of his work across various domains of mathematics and science.

When Gauss was a young boy, around the age of 7 or 8, he was in a classroom in Brunswick (now in Germany). His teacher, hoping to keep the class occupied, assigned the students a tedious task: to add up all the numbers from 1 to 100. The teacher expected this to take a while, giving him some time to relax.

However, Gauss quickly found a clever solution. Instead of adding the numbers sequentially, he noticed a pattern. He realized that if you pair the numbers from opposite ends of the sequence, each pair sums to the same value:

$$1 + 100 = 101$$
$$2 + 99 = 101$$
$$3 + 98 = 101$$
$$\vdots$$

Gauss saw that there were 50 such pairs (from 1 to 100), and each pair summed to 101. Therefore, the sum of the numbers from 1 to 100 could be calculated as:

$$\sum_{i=1}^{100} i = 50 \times 101 = 5050$$

But the teacher didn't know how he computed it so quickly; help the teacher understand Gauss's method.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^5$) — the number of test cases. Then follows their descriptions.
The single line of each test case contains one non-negative integer **n** ($1 \leq n \leq 46340$).

Output

For each test case, output the sum from 1 to n.

$$sum = \sum_{i=1}^{n} i$$

Standard Input	Standard Output
3 100 50 10	5050 1275 55

Explanation

In the first test case $n = 100$

$$\sum_{i=1}^{100} i = 1 + 2 + 3 + ... + 99 + 100 = 5050$$

Int the second test case $n = 50$

$$\sum_{i=1}^{50} i = 1 + 2 + 3 + ... + 49 + 50 = 1275$$

Note

Don't forget to print new line after every test case.

D. The Name Enigma

Input file: Standard Input
Output file: Standard Output
Time limit: 2 Seconds
Memory limit: 512 Megabytes

Raufun Nasim Khan Choudhury and ***Robiul Islam Rony***, the newly elected President and Vice President of ***CISC***, had a conversation about the length of their names. They were curious to know how many distinct characters are used in their full names.
So, write a program to help our President and Vice President to solve the problem.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^3$) — the number of test cases. Then follows their descriptions.
The first line of each test case contains one non-negative integer n ($1 \leq n \leq 10^5$) length of the string.
The second line of each test case contains a string in lowercase characters.

Output

For each test case, output a single integer representing the number of distinct characters in the given string.

Standard Input	Standard Output
2 24 raufunnasimkhanchoudhury 15 robiulislamrony	14 11

In the first test case "raufunnasimkhanchoudhury":The distinct characters are: 'r', 'a', 'u', 'f', 'n', 's', 'i', 'm', 'k', 'h', 'c', 'o', 'd', 'y'.
So, there are 14 distinct characters in this string.

E. Player Distribution

Input file: Standard Input
Output file: Standard Output
Time limit: 1 Seconds
Memory limit: 512 Megabytes

Raul Arvin is the football team captain of **Northern University of Business and Technology Khulna**. Every team has both good and bad players. During practice matches, not all players attend. Raul lists every player along with a skill rating, like 1, 2, 3, 4, and so on. In practice matches, he faces the challenging task of dividing the attended players into two teams based on their skill ratings.
He aims to minimize the skill rating difference between the two teams. Help him determine the **minimum** skill rating difference possible between the two teams.

Input

The first input line has an integer $n(1 \leq n \leq 20)$: the number of player. The next line has n integers $p_1, p_2, \ldots, p_n(1 \leq p_i \leq 10^9)$: the skill rating of each player.

Output

Print one integer: the **minimum** skill rating difference possible between the two teams

Standard Input	**Standard Output**
4 1 2 3 4	0

Standard Input	**Standard Output**
5 3 2 7 4 1	1

Explanation

In the first table $n = 4$
If we select the player:
First Team = 1 + 4 = 5
Second Team = 2 + 3 = 5
Minimum difference = First Team - Second Team = 5 - 5 = 0

In the second table $n = 5$
If we select the player:
First Team = 3 + 2 + 4 = 9
Second Team = 7 + 1 = 8
Minimum difference = First Team - Second Team = 9 - 8 = 1

Note: In some scenario, one can play against many players according to their skill level.
For example, **Messi or Ronaldo** can play against many players.

F. Divisor Count

Input file: Standard Input
Output file: Standard Output
Time limit: 1 Seconds
Memory limit: 512 Megabytes

Jeba is obsessed with number theory. She likes to find the properties of numbers, such as whether a number is prime or not, and how many divisors it has. Nowadays, she is working on finding the divisors of a number. She already knows that a number **n** has two common divisors: 1 and the **n** itself. She wants to know how many divisors a number **n** has, excluding 1 and the **n** itself.
Write a program to help Jeba with this task.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^3$) — the number of test cases. Then follows their descriptions.
The single line of each test case contains one non-negative integer **n** ($1 \leq n \leq 10^9$).

Output

For each test case, output a single integer representing the total number of divisors a number n has, excluding 1 and the n itself.

Standard Input	**Standard Output**
3	2
10	1
25	7
100	

In the first test case 10 has 4 divisor 1, 2, 5, 10. But jeba want's know excluding 1 and the 10 itself. So it has 2 valid divisor on this constraint's 2, 5.

G. Classical Information

Input file: Standard Input
Output file: Standard Output
Time limit: 1 Seconds
Memory limit: 512 Megabytes

Classical information refers to information that is processed, stored, and transmitted using classical physics principles.The basic unit of classical information is a bit. A bit can be in one of two classical states, typically represented as 0 or 1.The classical computer computes using these two possible classical states. Let us use the symbol Σ to refer to the set of classical states of our classical computer.

$$\Sigma = \{0, 1\}$$

All bits are in the classical states 0 or 1; they can't be in probabilistic states. We can use **Dirac notation** to represent classical information:
Classical States:
$|\rangle$ known as **ket** and it can be represent as matrix:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$\langle|$ known as **bra** and it can be represent as matrix:

$$\langle 0| = \begin{pmatrix} 1 \, 0 \end{pmatrix} \quad \langle 1| = \begin{pmatrix} 0 \, 1 \end{pmatrix}$$

Probabilistic state:
If we measure a system when it is in a probabilistic state, say for our classical computer before measuring any bit, it is in a probabilistic state. Let's say the probability of it being 0 is $\frac{3}{4}$ and the probability of it being 1 is $\frac{1}{4}$.
It can be represent as:

$$\frac{3}{4}|0\rangle + \frac{1}{4}|1\rangle = \begin{pmatrix} \frac{3}{4} \\ \frac{1}{4} \end{pmatrix}$$

Deterministic Operations:

An operation is deterministic if the result is completely determined by the input without any element of chance or uncertainty.
First, there are deterministic operations, where each classical state $a \in \Sigma$ is transformed into $f(a)$ for some function f of the form $f : \Sigma \to \Sigma$.

For example, if $\Sigma = \{0, 1\}$, there are four functions of this form, f_1, f_2, f_3, and f_4, which can be represented by tables of values as follows:

a	$f_1(a)$
0	0
1	0

a	$f_2(a)$
0	0
1	1

a	$f_3(a)$
0	1
1	0

a	$f_4(a)$
0	1
1	1

The actions of deterministic operations on probabilistic states can be represented by matrix-vector multiplication. Specifically, the matrix M that represents a given function $f : \Sigma \to \Sigma$ is the one that satisfies

$$M|a\rangle = |f(a)\rangle$$

for every $a \in \Sigma$. Such a matrix M always exists and is unique.

For any function $f : \Sigma \to \Sigma$, we may express the matrix M corresponding to the function f as

$$M = \sum_{a \in \Sigma} |\ f(a)\rangle\langle a\ | \ .$$

Now, if we again think about vectors as matrices, but this time consider the multiplication $\langle a\ |\rangle$, then we obtain a 1×1 matrix, which we can think about as a scalar. For the sake of tidiness, we write this product as $\langle a \mid b\rangle$ rather than $\langle a \,||\, b\rangle$. This product satisfies the following simple formula:

$$\langle a \mid b\rangle = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{if } a \neq b. \end{cases}$$

Using this observation, together with the fact that matrix multiplication is associative and linear, we obtain

$$M|b\rangle = \left(\sum_{a \in \Sigma} |f(a)\rangle\langle a|\right)|b\rangle = \sum_{a \in \Sigma} |f(a)\rangle\langle a \mid b\rangle = |f(b)\rangle$$

for each $b \in \Sigma$, which is what we require of M.
Write a program to find a 2×2 matrix M.

Input

It consists of a 2×2 matrix. The first column of that matrix, a ($0 \leq a \leq 1$), represents the input of the function f, and the second column of that matrix is the output of that function $f(a)$ ($0 \leq f(a) \leq 1$).

Output

Print the 2×2 matrix M.

Standard Input	**Standard Output**
0 0 1 0	1 1 0 0

Explanation

$$\begin{array}{c|c} a & f(a) \\ \hline 0 & 0 \\ 1 & 0 \end{array}$$

From the table(input):
In the first row : $a = 0$, $f(a) = 0$
The output matrix:

$$M = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$$

According to the equation:

$$M|a\rangle = |f(a)\rangle$$

$$M|0\rangle = |0\rangle$$

Left hand side:

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \times 1 + 1 \times 0 \\ 0 \times 1 + 0 \times 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle = |f(a)\rangle$$

In the second row: $a = 1,\ f(a) = 0$
According to the equation:

$$M|a\rangle = |f(a)\rangle$$
$$M|1\rangle = |0\rangle$$

Left hand side:

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \times 0 + 1 \times 1 \\ 0 \times 0 + 0 \times 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle = |f(a)\rangle$$

Now we can say that there exist a matrix M.

$$M = \sum_{a \in \Sigma} |\ f(a)\rangle\langle a\ | \ .$$

we know that a and $f(a)$ can be 0 or 1.

$$\Sigma = \{0, 1\}$$

From the table(input):
In the first row : $a = 0$, $f(a) = 0$
In the second row: $a = 1,\ f(a) = 0$

$M = \sum_{a \in \Sigma} |\ f(a)\rangle\langle a\ |$

$M =|\ 0\rangle\langle 0\ | + |\ 0\rangle\langle 1\ |$

$M = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix}$

$M = \begin{pmatrix} 1 \times 1 & 1 \times 0 \\ 0 \times 1 & 0 \times 0 \end{pmatrix} + \begin{pmatrix} 1 \times 0 & 1 \times 1 \\ 0 \times 0 & 0 \times 1 \end{pmatrix}$

$M = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$

$M = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$