

## Question - 1

### Metals

Mr. Octopus has recently shut down his factory and wants to sell off his metal rods to a local businessman. Mr. Octopus has many rods whose length are represented by array - `lengths = {lengths[0], lengths[1], lengths[2], ...}`.

The local businessman will only pay for rods that have same length. Let's say Mr. Octopus plans to sell rods of length `L` only. Then he had to cut each rod 0 or more time, so that he can maximize the profit. The remaining rods whose length is not `L` will be thrown away. Price of `N` rods of length `L` will be  $N \times L \times \text{unit\_price}$ . Also note that for each cut made to a rod, he had to pay `cut_cost`.

What is the maximum amount of money Mr. Octopus can make? You have to complete the function `int maxProfit(int cut_cost, int unit_price, int[] lengths)`.

Constraints:

- `lengths` will contain between 1 to 50 elements, inclusive.
- Each element of `lengths` will lie in range `[1, 10,000]`.
- $1 \leq \text{unit\_price}, \text{cut\_cost} \leq 1,000$ .

Input Format:

The function "maxProfit" contains three arguments (`int cut_cost, int unit_price, int[] lengths`), `cut_cost`, `unit_price`, `length` array respectively.

Output Format:

Return an integer denoting the maximum profit that can be gained by Mr. Octopus.

Sample Input #00:

```
cut_cost = 1
unit_price = 10
lengths = [26, 103, 59]
```

Sample Output #00:

```
1770
```

Explanation #00:

Since cuts are pretty cheap, we can make large number of cuts to reduce the amount of waste. The optimal length of rods will be 6. We can cut 4 pieces of length 6 from 1st rod, and throw piece of length 2, then cut 17 pieces of length 6 from 2nd rod and throw away a piece of length 1. From the third rod, we cut 9 pieces of length 6 and throw a piece of length 5. So in total we have 30 pieces of length 6 and we have made 30 cuts also. So total profit is  $30 \times 6 \times 10 - 30 \times 1 = 1770$

Sample Input #01:  
cut\_cost = 100  
unit\_price = 10  
lengths = [26, 103, 59]

Sample Output #01:  
1230

Explanation #01:  
Here we will throw smallest rod entirely and cut the pieces of length 51 from both left. So profit is  $3 \times 51 \times 10 - 3 \times 100 = 1230$ .

## Question - 2

### Anagrams

---

Two strings AA and BB are called anagrams if they consist same characters, but may be in different orders. So the list of anagrams of CAT are "CAT", "ACT", "TAC", "TCA", "ATC" and "CTA".  
Given two strings, print "Anagrams" if they are anagrams, print "Not Anagrams" if they are not. The strings may consist at most 50 english characters, the comparison should NOT be case sensitive, **but should ignore spaces if any**.  
Input is in the form of pair of strings separated by a '-'.  
Create a java program to read two set of Inputs as shown below.

Sample i/p:  
**hardware-aaedwhi**  
**harvard-dar vrah**

Sample o/p:  
**Not Anagrams**  
**Anagrams**

## Question - 3

### Fantabulous Arrays

---

A fantastic array is one in which each new element in the array is greater than the sum of all the previous elements. For example, { 2, 3, 6, 13 } is a fantastic array while { 2, 3, 5, 11 } is not a fantastic array.

A fabulous array is one in which each new element in the array is greater than the product of all the previous elements. For example, { 2, 3, 9, 88 } is a fabulous array while { 2, 3, 6, 54 } is not a fabulous array.

A array is said to be fantabulous if it is both fantastic and fabulous.

Write a Java program to check if a given array of integers is fantastic, fabulous or fantabulous. Your program should output one of the below according to the

type of the array:

- Fantastic Array
- Fabulous Array
- Fantabulous Array
- Array (if the array is none of the above)

## Question - 4

Trailing Zeros

---

Write a program to find the number of trailing zeros in the factorial of a given integer n.

Input Format:

There will be one line of input, with one integer n.

Output Format:

Print the count of trailing zeros.

Constraints:

$0 \leq n \leq 10000$

Sample Input #00:

23

Sample Output #00:

4

Explanation #00:

The factorial of 23 is: **25,852,016,738,884,976,640,000**

This value has four trailing zeros (at the end).

## Question - 5

Merge and sort arrays

---

create a program to read two integer arrays as input and merge the two arrays , sort , and print the new array contents

the format for input and output are given below.

Sample i/p

3

4 10 2

5

8 5 93 10 2

Sample o/p: 2 2 4 5 8 10 10 93

## Question - 6

Quality of Steel

---

A certain Steel plant categorizes its Steel using the following conditions:

- Hardness must be greater than 50
- Carbon content must be less than 0.7
- Tensile strength must be greater than 5600

The Steel is assigned grades in the following manner:

- Grade is 10 if all three conditions are met.
- Grade is 9 if conditions (i) and (ii) are met.
- Grade is 8 if conditions (ii) and (iii) are met.
- Grade is 7 if conditions (i) and (iii) are met.
- Grade is 6 if only one condition is met.
- Grade is 5 if none of three conditions are met.

Write a Java program which outputs the grade of the Steel for the given values of hardness, carbon content and tensile strength.

## Question - 7

### Moody Mode

---

Mode is defined as the number which occurs most frequently in a given dataset. Given a set of five numbers 13, 11, 23 13, 21, the mode would be 13 as it occurs twice and the rest of the numbers occur only once. This challenge is to write a program to find the mode of a given list of integers. If there is one clear mode, print that number, if there are more than one modes, print all the numbers in sorted order, if there is no mode, print "Moody Mode".

examples:

**I/P:**

23 19 2 4 23 4 23

**O/P:**

23

(23 repeats 3 times)

**I/P:**

23 19 2 4 2 19

**O/P**

2

19

**I/P:**

1 2 3 4 5 6

**O/P:**

Moody Mode

## Question - 8

### Uneaten Leaves

---

K caterpillars are eating their way through N leaves. Each caterpillar falls from leaf to leaf in a unique sequence. All caterpillars start at a twig in position 0 and fall onto the leaves at positions between 1 and N. Each caterpillar i has an associated 'jump-number'  $A_i$ . A caterpillar with jump number j eats leaves at positions that are multiples of j. It will proceed in the order j, 2j, 3j, ... till it reaches the end of the leaves, then it stops and builds its cocoon.

Given a set A of K elements, we need to determine the number of uneaten leaves.

**Input Format:**

N = number of leaves

K = number of caterpillars

A = Array of integer jump numbers (one per line).

**Constraints:**

- $1 \leq N \leq 2 \times 10^9$
- $1 \leq K \leq 18$
- $2 \leq A[i] \leq 22$

**Output Format:**

Print an integer denoting the number of uneaten leaves.

**Sample Input 0**

```
10
3
2
4
5
```

**Sample Output 0**

```
4
```

**Explanation**

All leaves which are multiples of 2, 4 and 5 are eaten.

Only 4 leaves which are numbered 1, 3, 7 and 9 are left.

**Sample Input 1**

```
20
3
3
7
5
```

**Sample Output 1**

```
9
```

**Explanation**

All leaves which are multiples of 3, 5 and 7 are eaten.

Only 9 leaves which are numbered 1, 2, 4, 8, 11, 13, 16, 17 and 19 are left.

**Question - 9**

Sum the digits

---

Create a Java Program to find Sum of Digits of a number N, given as input.

Range of N is as below.

**0<=N <= 9223372036854775807**

Sample i/p : 1234

Sample o/p:10

**Question - 10**

Count the number of alphabets in the sentence

---

Count and print the number of alphabets in a sentence.

sample ip: How are you ?

Sample o/p: 9