The Protolympics

Easy (5 points each)

1. **Late Doctors:** Doctors are always late. You're given a doctor appointment at time **Ta**. Doctor arrives at time **Td**. You are willing to wait for **Tw** hours. If the doctor arrives after **Tw**, he is a late doctor (typical doctor). If not, he is a good doctor 🙂

   <u>Input:</u>   Ta Td Tw   <u>Output:</u> Good doctor/Late doctor

   3 5 1   Late Doctor

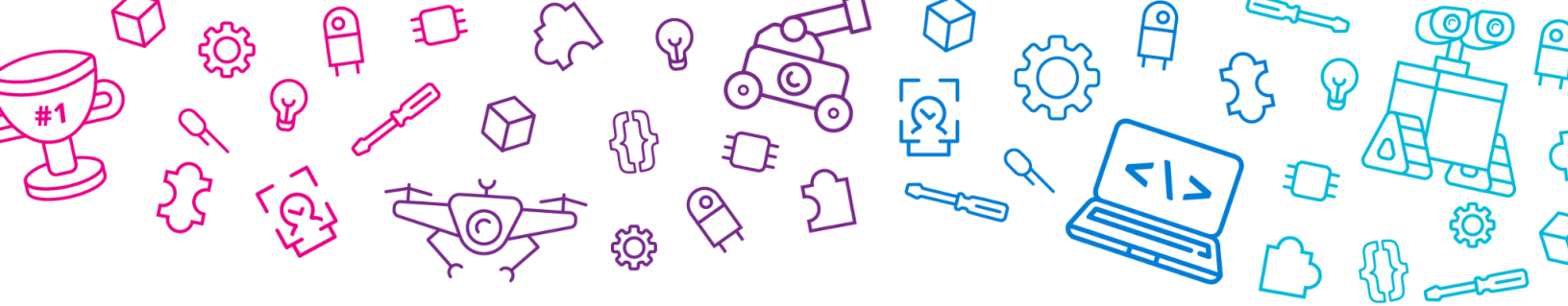   2 9 8   Good Doctor

2. **Frame Printer:** You work at a printing store. Your clients give you a couple of words in a list and you want to print them in a nicely made frame. For example the list `["Hello", "World", "frame"]` gets printed as:

   ```
   ********
   * Hello *
   * World *
   * frame *
   ********
   ```

   Note: All strings are guaranteed to be the same length.

3. **Space Numbers:** An Armstrong number is a number that is equal to the sum of its digits, each raised to the power of the number of digits. For example 1634 is an Armstrong number because:
$1^4 + 6^4 + 3^4 + 4^4 = 1634$
Take an integer and print if it's an Armstrong number or not.

4. **Anti-vowels:** Ziyad doesn't like vowel letters (a, e, i, o, u). So he's going to give you a **string**, and he needs you to remove all the vowels from it. Moreover, he wants all the capital letters replaced with their small letters counterparts.

   Input: Output:

   Hello World   hll wrld

5. **Counter:** Ziyad wants to count numbers, however his memory isn't good enough. So he needs you to count for him. Write a program that takes **n** lines, that are either "count up" which will increment or "count down" which will decrement. Given that you'll start counting from zero, print out the final count.

   Input: Output:

   5   2
   count up
   count down
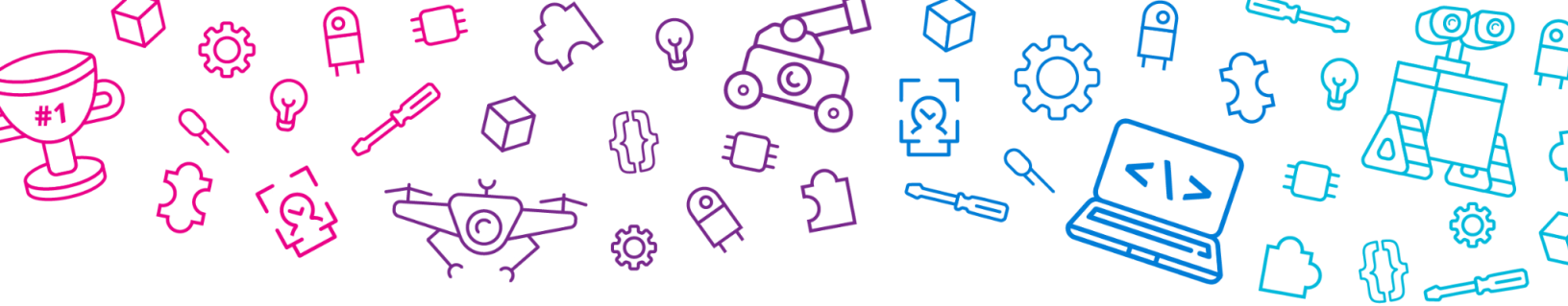   count up
   count up
   count up

6. **Watermelons:** Ziyad has now become Baba Ziyad, and every week in the summer he brings along a huge watermelon on his way home from work for his two kids. He's going to split the watermelon into 2 parts for each kid. His two kids don't like odd numbers, so he wants the weight of each part to be even. Given the total weight of the watermelon, find if it can be divided into 2 even-weighted parts or not.
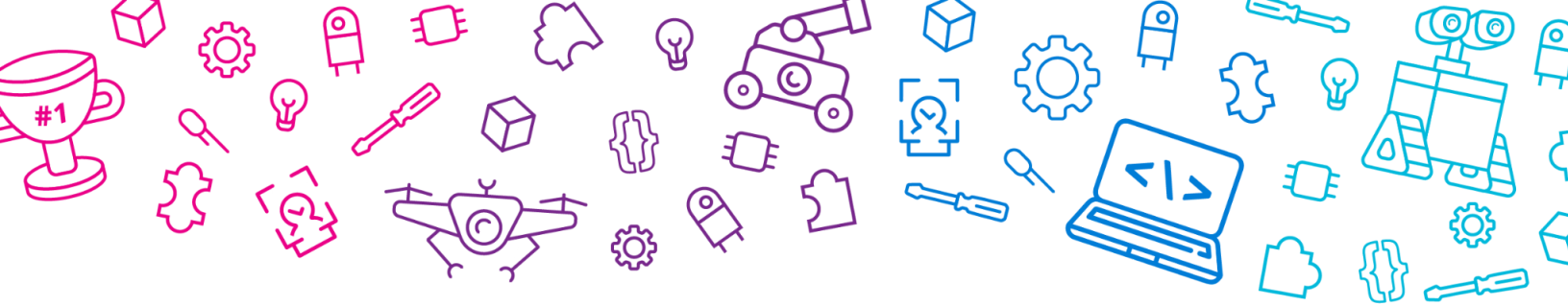
<u>Input:</u>

8

7

<u>Output:</u>

Yes

No

Medium (10 points each)

1. **Range Summer:** You'll take in **_n_** groups of numbers, represented as ranges (a number as the start and another as the end). Your job is to calculate the sum of the numbers between the ranges.

   <u>Input:</u>   <u>Output:</u>

   2   21

   1 3

   4 6

   <u>Explanation:</u> First line of input contains **_n_**, which is 2. Then 2 groups are entered, 1 to 3 and 4 to 6. 1st sum is 1+2+3 = 6 and 2nd sum is 4+5+6 = 15, and 6+15 = 21

2. **Shampoo Problem:** A family consists of 3 people (mother, father, and son), and each of them uses **_X_** ml of shampoo each day. You'll take the size of the shampoo bottle, and then how many each family member uses each day(size,mother,father,son). Your task is to figure out who'll run out of shampoo first.

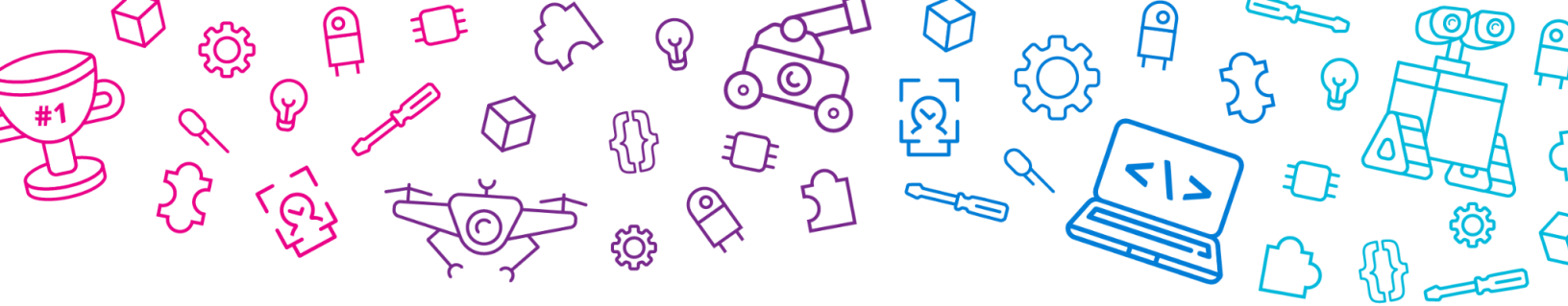   <u>Input:</u>   <u>Output:</u>

   30, 13, 11, 12   son

   30, 10, 10, 10   mother

Case 1 Explanation: The shampoo bottle is the first number (30), the mother will use 13, leaving 17, then the father uses 11 leaving 6, and so the son can't shower.

Case 2 Explanation: The whole family will shower on the 1st day, but on the 2nd it will be finished so mother won't be able to shower.

3. **On the Same Range:** Ahmad likes numbers from *A* to *B*, while Mahmoud likes numbers from *C* to *D*. Is there any chance that there's a number they both like? Input format is *A B C D*.

Input: Output:

10 20 30 40   No

10 30 20 40   Yes

4. **Sort of a Problem:** Check if a list is sorted or not, but **do not use any built-in functions**.

Input: Output:
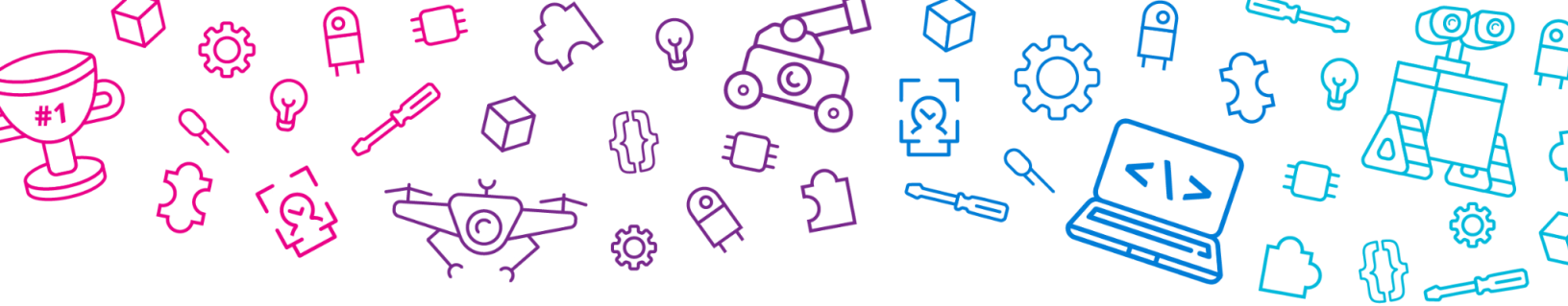
1, 3, 2, 4, 5, 6   No

5, 6, 7   Yes

5. **Palindrome Buster:** You're given a list of strings, print the indices of the strings that are palindromes.

Input: Output:

car tacocat ball   1

racecar food sun pop   0 3

6. **Reverse Fibonacci:** You'll take in a number from the Fibonacci sequence, your job is to return its index in the series.

Fibonacci Series: 1 1 2 3 5 8 13 21 34 …

Input: Output:

55   10

13   8

7. **Pangram Program:** Check if a sentence is a pangram or not. A pangram is a sentence that contains all of the letters of the alphabet at least once.
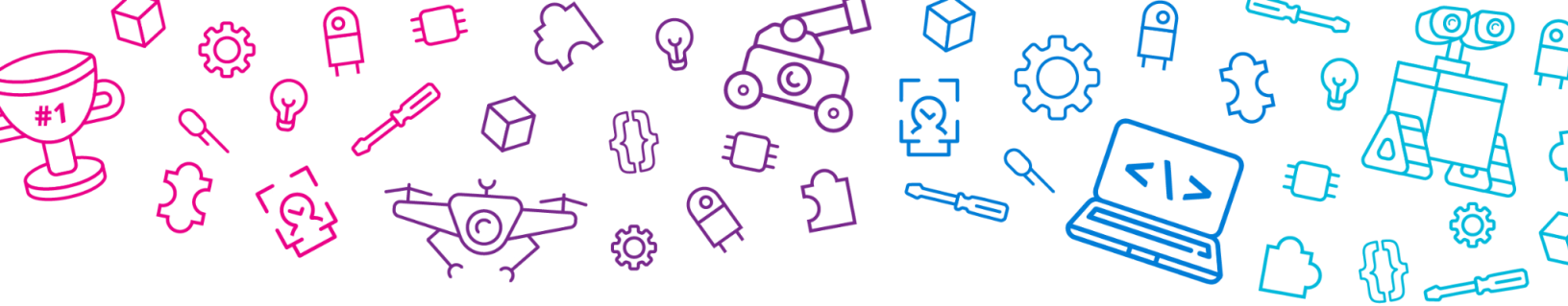
Input: Output:

The quick brown fox jumps over a lazy dog   Yes

I love protons   No

PROTONS

8. **Perfect Splitter:** Given a ***string*** of length ***n***, you'll also take a number ***k*** which is a factor of ***n***. Your job is to divide the string into n/k substrings and eliminate the repetitions in each one. Input is ***string***, ***k***.
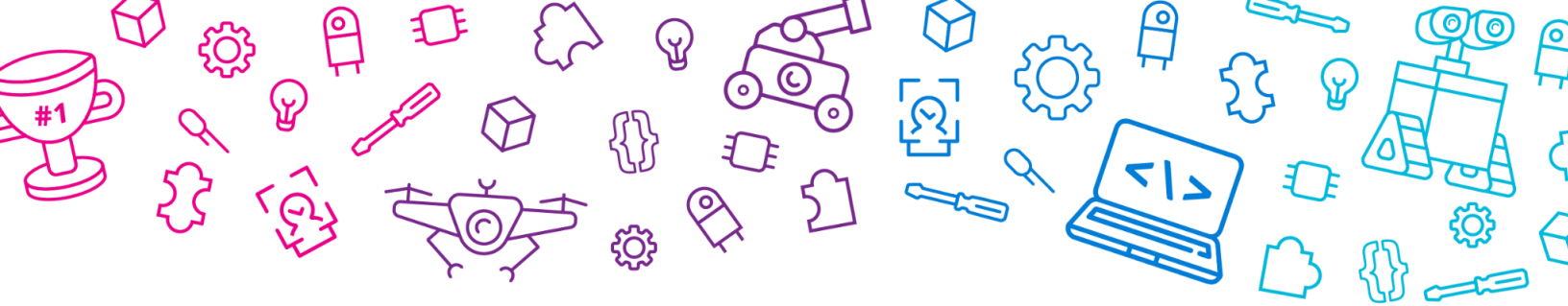
Input: Output:

AAABCADDE 3   ABCADE

ABBCCCDF 4   ABCCDF

Hard (20 points each)

1. **Sine Here Please:** The sine of an angle **x** can be approximated by calculating the result of the first **N** terms of the series:

$$sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + ... \ (x \text{ in radians})$$

You'll take in the angle **x** and the number of terms **N** to use.

Input:	Output:
30, 100	0.5
40, 3	0.64
45, 10	0.71

Note: In Python's "math" library, there's a "factorial" function, you could use it here.

2. **Comma Geek:** Create a function that formats numbers so that between each 3 digits there is a comma (",").
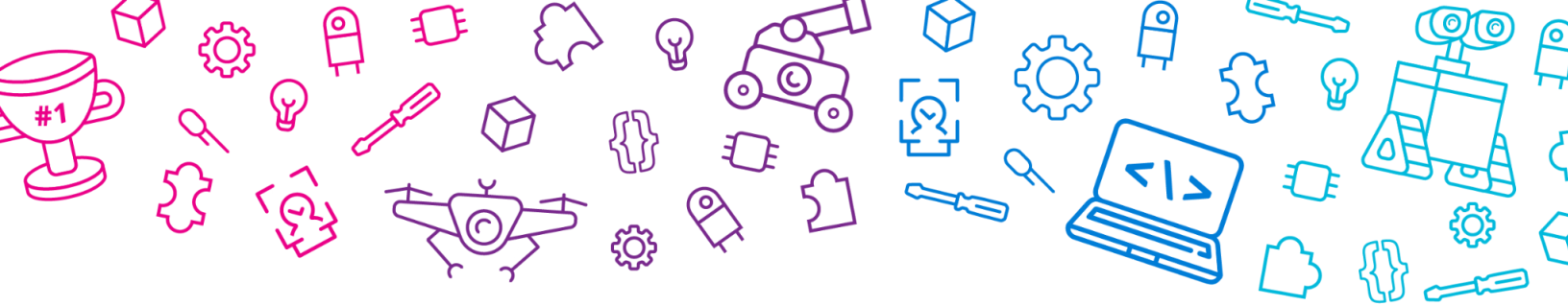
Input:	Output:
100000	10,000
12345678	12,345,678

3. **Time Master:** You'll take the **_time_** in the 24 hour format, and your job is to find out what the time will be after **_x_** minutes. Input format is **_time_**, **_x_**.

Input: Output:

23:59, 10   after 10 mins it'll be 00:09

20:20, 121   after 121 mins it'll be 22:21

4. **Combo Number:** You'll take in a number **_X_**, and your job is to figure out how many numbers(A, B, and C) can satisfy the equation **"X = A * B + C"**.

Note: A,B and C must be positive integers

Input: Output:

3   3

100   473

Case 1 Explanation: Possible combinations are (1,1,2),(1,2,1),(2,1,1)

5. **Summer Busting Numbers:** Given a string composed of letters and numbers, calculate the sum of all the numbers in the string.
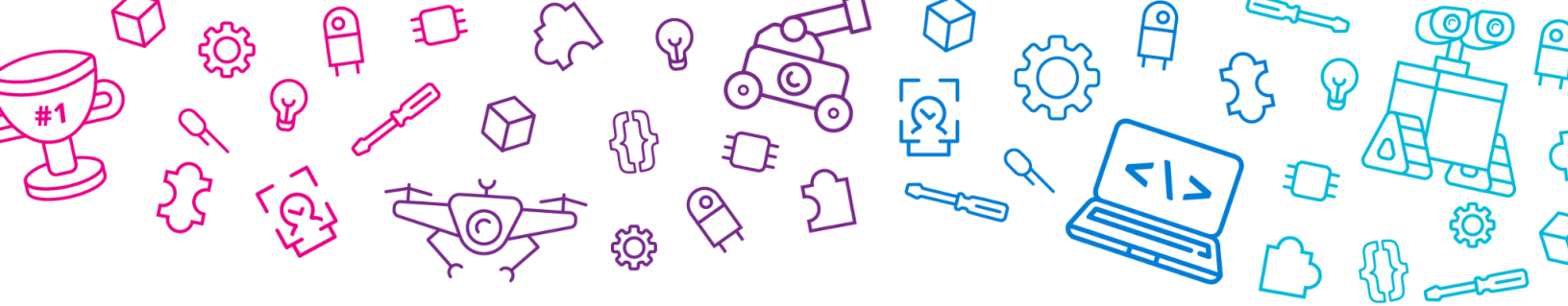
Input: Output:

a12b34cd   46   _(12 + 34)_

a0bc123   123   _(0 + 123)_

6. **Binary Guru:** You'll take in a string of 1's and 0's. In one step you can eliminate "10" or "01" from the string, and you have unlimited steps. Your task is to print out the length of the string after removing all the "01" and "10"'s you can.

Input:	Output:	
1100	0	*(remove middle 10 then 10)*
01010	1	*(remove 1st 01 then 2nd 01)*
11101111	6	*(remove 10 or 01)*