1. Create a function that takes a number n (integer greater than zero) as an argument, and returns 2 if n is odd and 8 if n is even.

You can only use the following arithmetic operators: addition of numbers +, subtraction of numbers -, multiplication of number *, division of number /, and exponentiation **.

You are not allowed to use any other methods in this challenge (i.e. no if statements, comparison operators, etc).

Examples

f(1) 2

f(2) 8

f(3) 2

2. Create a function that returns the majority vote in a list. A majority vote is an element that occurs > N/2 times in a list (where N is the length of the list).

Examples

majority_vote(["A", "A", "B"]) "A"

majority_vote(["A", "A", "A", "B", "C", "A"]) "A"

majority_vote(["A", "B", "B", "A", "C", "C"]) None

3. Create a function that takes a string txt and censors any word from a given list lst. The text removed must be replaced by the given character char.

Examples

censor_string("Today is a Wednesday!", ["Today", "a"], "-") "----- is - Wednesday!"

censor_string("The cow jumped over the moon.", ["cow", "over"], "*"), "The *** jumped **** the moon.")

censor_string("Why did the chicken cross the road?", ["Did", "chicken", "road"], "*") "Why *** the ******* cross the ****?"

4. In mathematics a Polydivisible Number (or magic number) is a number in a given number base with digits abcde... that has the following properties:

- Its first digit a is not 0.
- The number formed by its first two digits ab is a multiple of 2.
- The number formed by its first three digits abc is a multiple of 3.
- The number formed by its first four digits abcd is a multiple of 4.

Create a function which takes an integer n and returns True if the given number is a Polydivisible Number and False otherwise.

Examples

is_polydivisible(1232) True
1 / 1 = 1
12 / 2 = 6
123 / 3 = 41
1232 / 4 = 308

is_polydivisible(123220) False
1 / 1 = 1
12 / 2 = 6
123 / 3 = 41
1232 / 4 = 308
12322 / 5 = 2464.4 # Not a Whole Number
123220 /6 = 220536.333... # Not a Whole Number

5. Create a function that takes a list of numbers and returns the sum of all prime numbers in the list.

Examples

sum_primes([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]) 17

sum_primes([2, 3, 4, 11, 20, 50, 71]) 87

sum_primes([]) None