ASSIGNMENT FUNCTIONS PROBLEMS

Q1. Write a function **check_even_odd(n: int)** that determines if a given integer n is even or odd and prints the result.

Input:
An integer n.
Output:
Print "even" if n is even.
Print "odd" if n is odd.

Ask input from user and then pass it as arguments to a function.

Q2. Write a function **check_divisibility(a: int, b: int)** that checks if the first integer **a** is divisible by the second integer **b** and prints the result.

Input:
Two integers a and b.
Output:
Print **True** if a is divisible by b.
Print **False** if a is not divisible by b.

Ask input from user and then pass it as arguments to a function.

Example
a = 10
b = 5
True

Q3. Create a function named **simple_calculator** that takes three parameters: **two integer number and an operation (string)** (addition or subtraction represented by '+' or '-'), and prints the result of the operation.

Ask input from user and then pass it to function.

Example
simple_calculator(15, 20, ”+”)
Output
35

Example
simple_calculator(-50, 50, ”+”)
Output
0

Example
simple_calculator(100, 20, ”-”)
Output
80

Q4. Create a function that takes **three integers as parameters** and prints the largest among them. You should ask 3 integers from user and then pass it to function.

Example 1
10
50
30
Output
50

Example 2
50
50
50
Output
50

Q5. In this challenge, establish if a given integer num is a C**urzon number**. If **1** plus **2** elevated to **num** is exactly divisible by **1** plus **2** multiplied by **num**, then **num** is a Curzon number.

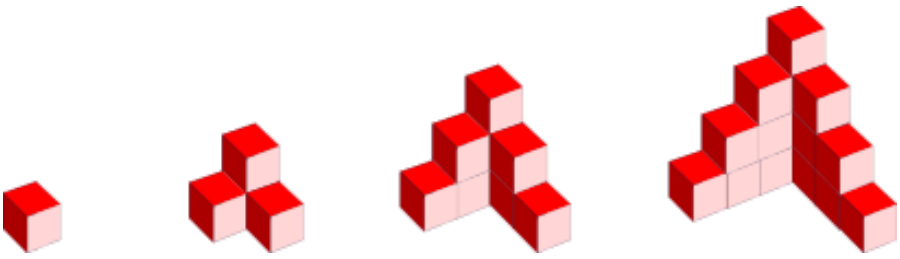
Given a non-negative integer **num**, implement a function that prints **True** if **num** is a Curzon number, or **False** otherwise.

Examples
is_curzon(5) → True
2 ** 5 + 1 = 33
2 * 5 + 1 = 11
33 is a multiple of 11

is_curzon(10) → False
2 ** 10 + 1 = 1025
2 * 10 + 1 = 21
1025 is not a multiple of 21

is_curzon(14) → True
2 ** 14 + 1 = 16385
2 * 14 + 1 = 29
16385 is a multiple of 29

Q6. Here's an image of four models. Some of the cubes are hidden behind other cubes. Model one consists of one cube. Model two consists of four cubes, and so on...



Write a function that takes a number **n** and prints the number of stacked boxes in a model **n** levels high, visible and invisible.

Ask **n** from user.

Examples

stack_boxes(1) → 1

stack_boxes(2) → 4

stack_boxes(0) ➞ 0

Q7. For each of the 6 coffee cups I buy, I get a 7th cup free. In total, I get 7 cups. Create a function that takes **n** cups bought and print the total number of cups I would get.

Examples

total_cups(6) ➞ 7

total_cups(12) ➞ 14

total_cups(213) ➞ 248

Notes

Number of cups I bought + number of cups I got for free.