1. Insert a number. Print ‘yes’ if the number is prime, ‘no’ otherwise.

Input	Output
401	‘yes’
63	‘no’

2. Given a number ***n (n>= 0)***. Print ***n***th Fibonacci number. *(Fibonacci series: 0, 1, 1, 2, 3, 5, 8 …,* $a_k = a_{k-1} + a_{k-2}$)

Input	Output
0	0
2	1
10	55
20	6765

3. Given a number ***n(n> 0)***. Print Fibonacci series up to ***n***.

Input	Output
7’	“0, 1, 1, 2, 3, 5 “
45	“0, 1, 1, 2, 3, 5, 8, 13, 21, 34”

4. Insert a number. Calculate product and sum of the digits of the number. If product is divisible by the sum, print the quotient, otherwise print the remainder.

Input	Output
1233	‘Quotient is 2.’
5	‘Quotient is 1.’
0	‘Cannot calculate.’
455	‘Remainder is 2.’

9. Given three numbers **a, b** (a ≤ b) and **num.** Create an array of evenly spaced numbers by the given **num** length over the specified interval (from **a** to **b).**

Input	Output
1 5 1	[1]
10 100 3	[10, 55, 100]
1 5 6	[1 , 1.8, 2.6, 3.4, 4.2, 5]

10. Given an array of numbers. Find the index of the second maximum element.

Input	Output
[23, -98, 0, -456, 12, 8]	4
[-60, 2, 43, -18, 5, -19, 36, 7, 56]	2

11. Given an array of numbers, padding amount and repeat count. Pad the array in the following way: the padding amount specifies how many elements should be taken from the array edges, the repeat amount specifies how many times the pad should be repeated. Also, you should check that *padding amount <= length of array.*

Input	Output
array = [1, 2, 3, 4] padAmount = 1 repeat = 3	[1, 1, 1, 1, 2, 3, 4, 4, 4, 4]
padAmount = 2 repeat = 1	[1, 2, 1, 2, 3, 4, 3, 4]
array = [1] padAmount = 1 repeat = 3	[1, 1, 1, 1, 1, 1, 1]
array = [1] padAmount = 2 repeat = 3	“Invalid padding amount”