

Conditions

1. Odd or Even:

Write a program that takes a positive integer n as input and prints whether it is odd or even.

Sample Input 1:

49

Sample Output 1:

Odd

Sample Input 2:

36

Sample Output 2:

Even

2. Grading System:

Write a program that takes a student's score (out of 100) and outputs their grade based on the following conditions:

- 90-100: Grade A
- 80-89: Grade B
- 70-79: Grade C
- 60-69: Grade D
- Below 60: Grade F

Sample Input 1:

78

Sample Output 1:

C

Sample Input 2:

90

Sample Output 2:

A

Sample Input 3:

48

Sample Output 3:

F

3. Age Group Categorizer

Write a program that categorizes a person's age into different life stages based on the following conditions:

- 0-2 years: Infant
- 3-12 years: Child
- 13-19 years: Teenager
- 20-65 years: Adult
- Above 65 years: Senior

The program takes an age as input and prints the age group category the person belongs to.

Sample Input 1:

15

Sample Output 1:

Teenager

Sample Input 2:

2

Sample Output 2:

Infant

Sample Input 3:

72

Sample Output 3:

Senior

4. Temperature Adviser

Write a program that suggests an activity based on the current temperature (in Celsius), following these guidelines:

- Above 30°C: "It's hot. Let's go swimming!"
- 20°C to 30°C: "Perfect for a picnic."
- 10°C to 19°C: "Maybe wear a jacket?"
- Below 10°C: "Too cold! Best to stay indoors."

The program takes the current temperature as input and prints the suggested activity.

Sample Input 1:

15

Sample Output 1:

Maybe wear a jacket?

Sample Input 2:

8

Sample Output 2:

Too cold! Best to stay indoors.

Sample Input 3:

38

Sample Output 3:

It's hot. Let's go swimming!

5. Maximum of Three Numbers:

Write a program that takes three numbers as input and prints the largest number.

Sample Input 1:

15 23 16

Sample Output 1:

23

Sample Input 2:

8 5 9

Sample Output 2:

9

6. Sort Three Numbers

Write a program that takes three numbers as input and prints the numbers in ascending order.

This program doesn't use an array.

Sample Input 1:

15 23 16

Sample Output 1:

15 16 23

Sample Input 2:

8 5 9

Sample Output 2:

5 8 9

Loops

7. N times "Hello World":

You are given an integer n. Print "Hello World" n times.

Sample Input:

5

Sample Output:

Hello World

Hello World

Hello World

Hello World

Hello World

8. N times * in same line:

You are given an integer n. Print * n times in the same line with space in between.

Sample Input:

5

Sample Output:

* * * * *

9. First n Natural Numbers:

You are given an integer n. Print the first n natural numbers.

Sample Input:

7

Sample Output:

1 2 3 4 5 6 7

10. All Even Numbers till n:

You are given an integer n. Print all the even numbers less than equal to n.

Sample Input:

14

Sample Output:

2 4 6 8 10 12 14

11. All Square Numbers till n:

You are given an integer n. Print all the square numbers less than equal to n.

Sample Input:

50

Sample Output:

1 4 9 16 25 36 49

12. First n Even Numbers:

You are given an integer n. Print the first n even numbers.

Sample Input:

7

Sample Output:

2 4 6 8 10 12 14

13. First n Odd Numbers:

You are given an integer n. Print the first n odd numbers.

Sample Input:

7

Sample Output:

1 3 5 7 9 11 13

14. First n Square Numbers:

You are given an integer n. Print the first n square numbers.

Sample Input:

7

Sample Output:

1 4 9 16 25 36 49

15. Multiplication Table of n:

You are given an integer n. Print the multiplication table of n till count 10.

Sample Input:

7

Sample Output:

7 14 21 28 35 42 49 56 63 70

Explanation: Print $7*1$, $7*2$, ..., $7*10$.

16. Factors of a Number:

Write a program that takes a number n as input and prints all the factors of the number.

Sample Input:

24

Sample Output:

1 2 3 4 6 8 12 24

Explanation: The factors of 24 are 1, 2, 3, 4, 6, 8, 12, 24.

17. Series: 3, 5, 7, 9, 11, ...:

You are given an integer n. Print first n terms of the series 3, 5, 7, 9, 11...

Sample Input:

7

Sample Output:

3 5 7 9 11 13 15

Explanation: The series starts with 3 and every time adds 2 to get the next term.

18. Series: 2, 5, 8, 11, 14, ...:

You are given an integer n. Print first n terms of the series 2, 5, 8, 11, 14...

Sample Input:

7

Sample Output:

2 5 8 11 14 17 20

Explanation: The series starts with 2 and every time adds 3 to get the next term.

19. Series: 3, 6, 12, 24, 48, ...:

You are given an integer n. Print first n terms of the series 3, 6, 12, 24, 48...

Sample Input:

7

Sample Output:

3 6 12 24 48 96 192

Explanation: The series starts with 3 and every time multiplies 2 to get the next term.

20. Sum of First n Natural Numbers:

You are given an integer n. Print the sum of the first n Natural Numbers.

Note: Use a loop instead of a mathematical formula.

Sample Input:

7

Sample Output:

28

Explanation: Sum of $1+2+3+4+5+6+7 = 28$

21. Sum of Series: 3, 5, 7, 9, 11, ...:

You are given an integer n. Print the sum of the first n terms of the series 3, 5, 7, 9, 11....

Note: Use a loop instead of a mathematical formula.

Sample Input:

7

Sample Output:

63

Explanation: Sum of $3+5+7+9+11+13+15 = 63$

22. Factorial of a Number:

Write a program to calculate the factorial of a given number n.

The factorial of a number n is the product of all positive integers less than or equal to n.

Sample Input:

6

Sample Output:

720

Explanation: $720 = 6*5*4*3*2*1$

23. Counting Digits in a Number:

Write a program that takes a number n as input and prints the number of digits the number has.

Sample Input:

1132

Sample Output:

4

24. Sum of Digits of a Number:

Write a program that calculates the sum of all the digits in a given number n.

Sample Input:

1132

Sample Output:

7

Explanation: $7 = 1+1+3+2$

Advanced Loops

25. Fibonacci Series:

Write a program that takes a number n as input and prints the first n terms of the Fibonacci Series.

The Fibonacci Series is a series of numbers where each number is the sum of the two preceding ones, usually starting with 0 and 1.

Series: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...

Sample Input:

10

Sample Output:

0 1 1 2 3 5 8 13 21 34

26. Reverse a Number:

Write a program that takes a number n as input and prints the reverse of the given number.

Sample Input:

1132

Sample Output:

2311

Explanation: The reverse of 1132 is 2311.

27. GCD (Greatest Common Divisor) or HCF (Highest Common Factor):

You are given two positive integers a and b . Print the GCD or HCF of these two numbers.

Sample Input:

20 16

Sample Output:

4

28. LCM (Least Common Multiple):

You are given two positive integers a and b . Print the LCM of these two numbers.

Sample Input:

20 16

Sample Output:

80

29. Binary to Decimal:

Write a program that takes a Binary Number B as input and prints the Decimal equivalent of the given input.

Sample Input:

1000

Sample Output:

8

30. Decimal to Binary:

Write a program that takes a Decimal Number D as input and prints the Binary equivalent of the given input.

Sample Input:

8

Sample Output:

1000

31. Is Prime?:

You are given an integer n. Print yes, if it is a prime number. Print no, if it is not a prime number.

Sample Input 1:

49

Sample Output 1:

No

Sample Input 2:

37

Sample Output 2:

Yes

32. Is Perfect Number?:

You are given an integer n. Print yes, if it is a Perfect Number. Print no, if it is not a Perfect Number.

A Perfect Number is a positive integer that is equal to the sum of its positive divisors, excluding the number itself.

Sample Input 1:

28

Sample Output 1:

Yes

Explanation 1: The factors of 28 are 1, 2, 4, 7, 14, 28. We don't consider the number itself. $28 = 1+2+4+7+14$.

Sample Input 2:

24

Sample Output 2:

No

Explanation 2: The factors of 24 are 1, 2, 3, 4, 6, 8, 12, 24. We don't consider the number itself. But $24 \neq 1+2+3+4+6+8+12$.

33. Is Armstrong Number?:

You are given an integer n. Print yes, if it is an Armstrong Number. Print no, if it is not an Armstrong Number.

An Armstrong Number is a positive number that equals the sum of its digits, each raised to the power of the number of digits.

Sample Input 1:

1634

Sample Output 1:

Yes

Explanation 1: 1634 has 4 digits, so we raise each digit to the power of 4. $1^4 + 6^4 + 3^4 + 4^4 = 1 + 1296 + 81 + 256 = 1634$

Sample Input 2:

153

Sample Output 2:

Yes

Explanation 2: 153 has 3 digits, so we raise each digit to the power of 3. $1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153$

34. Star Pattern 1:

Print the following pattern based on the given input.

Sample Input:

6

Sample Output:

```
*
**
***
****
*****
*****
```

Explanation: Since the input is 6, it prints a total of 6 lines. In each line, the star count increases.

35. Star Pattern 2:

Print the following pattern based on the given input.

Sample Input:

6

Sample Output:

```
  *
  **
 ***
****
*****
*****
```

Explanation: Since the input is 6, it prints a total of 6 lines. In each line, the star count increases and the leading space decreases.

36. Star Pattern 3:

Print the following pattern based on the given input.

Sample Input:

6

Sample Output:

```
*
 *
  *
   *
    *
     *
```

Explanation: Since the input is 6, it prints a total of 6 lines. In each line, it prints only one star, but the leading space increases.

37. Star Pattern 4:

Print the following pattern based on the given input.

Sample Input:

5

Sample Output:

```
  *
 ***
*****
*****
*****
```

Explanation: Since the input is 5, it prints a total of 5 lines of stars. Each line has stars as well as leading spaces. In each line, the star count increases by 2, but the leading space decreases by 1.

38. First n prime numbers:

You are given an integer n. Print the first n prime numbers.

Sample Input:

7

Sample Output:

2 3 5 7 11 13 17

Array Traversal

39. Square - Each Array Element:

You are given an integer array. Traverse through the array and print the square for each element.

Sample Input:

5 3 4 7 2 9

Sample Output:

25 9 16 49 4 81

40. Odd or Even - Each Array Element:

You are given an integer array. Traverse through the array and for each element, if the element is odd print "Odd", else print "Even".

Sample Input:

4 7 9 10 13 17

Sample Output:

Even

Odd

Odd

Even

Odd

Odd

41. Odd Count in an Array:

You are given an integer array. Print the number of odd numbers in the array.

Sample Input:

4 7 9 10 13 17

Sample Output:

4

Explanation: There are 4 odd numbers in the given numbers: 7, 9, 13, 17.

42. Number Count in an Array:

You are given an integer array. You are also given a number. Print the number of times the number appears in the array.

Sample Input:

4 7 9 10 7 14 12 4 7 27

7

Sample Output:

3

Explanation: The given number 7 appears 3 times in the given array.

43. Sum of Elements in an Array:

You are given an integer array. Add all the numbers present in the array and print the sum.

Sample Input:

10 5 6 3 4 3 5 6

Sample Output:

42

Explanation: $10+5+6+3+4+3+5+6 = 42$

44. Average of Elements in an Array:

You are given an integer array. Find the average of all the numbers present in the array.

Sample Input:

10 5 6 3 4 3 5 6

Sample Output:

5.25

Explanation: There are 8 elements in the given array. Sum = $10+5+6+3+4+3+5+6 = 42$. Average = $42/8 = 5.25$

45. Product of Elements in an Array:

You are given an integer array. Multiply all the numbers present in the array and print the final product.

Sample Input:

3 2 5 1 4

Sample Output:

120

Explanation: $3 * 2 * 5 * 1 * 4 = 120$

46. Minimum in an Array:

You are given an integer array. Find the minimum element of the array.

Sample Input:

10 5 6 3 4 3 5 6

Sample Output:

3

Explanation: Here 3 is the minimum number. But since 3 is present more than once, we print the index of the first occurrence.

47. Minimum Index in an Array:

You are given an integer array. Find the index of the minimum element of the array. If there are multiple occurrences of the minimum number, print the index of the first occurrence of the minimum number.

Note: The index will be calculated from 1.

Sample Input:

10 5 6 3 4 3 5 6

Sample Output:

4

Explanation: Here 3 is the minimum number. But since 3 is present more than once, we print the index of the first occurrence.

48. Maximum Sum of Two Consecutive Elements

You are given an integer array. Print the maximum sum of two consecutive integers in the given array.

Sample Input:

3 6 2 1 8 2 5 7 1

Sample Output:

12

Explanation: $5 + 7 = 12$ is the maximum sum of two consecutive integers in the given array.

49. Count Elements until Negative:

You are given an integer array. Count all the numbers present in the array till you encounter a negative number and print the count. If no negative number is present, count till the end.

Sample Input:

10 5 6 3 -1 4 -3 5 6

Sample Output:

4

Explanation: There are 4 elements before the first negative number appears.

50. Sum until Zero:

You are given an integer array. Add all the numbers present in the array till you encounter a 0 and print the sum. If no 0 is present, add till the end.

Sample Input:

10 5 6 3 0 4 3 5 6

Sample Output:

24

Explanation: $10+5+6+3 = 24$

51. Linear Search:

You are given an integer array. Take another number as input and search if the number is present in the given array. If the number is present, print "Yes", else print "No".

Sample Input 1:

11 1 13 21 3 7

3

Sample Output 1:

Yes

Sample Input 2:

11 1 13 21 3 7

5

Sample Output 2:

No

52. Check Array for Negative Numbers:

You are given an array of integers. Check if the array has any negative numbers. If the array has any negative number, print yes. Else, print no.

Sample Input 1:

11 1 13 21 3 7

Sample Output 1:

No

Explanation 1: The given array has no negative number.

Sample Input 2:

6 8 9 -1 14 8 -3 6

Sample Output 2:

Yes

Explanation 2: The given array has negative numbers, -1 and -3.

53. Check Array for Ascending Order:

You are given an array of integers. Check if the array is in Ascending Order. If yes, print "Yes", else print "No".

Sample Input 1:

3 5 10 13 16 12 14

Sample Output 1:

No

Explanation 1: The given array is not in ascending order.

Sample Input 2:

3 5 10 13 16 25 33

Sample Output 2:

Yes

Explanation 2: The given array is in ascending order.

54. First Perfect Square:

You are given an array of integers. Print the first element of the array that is a perfect square. If there are no perfect squares, print No.

Sample Input 1:

3 6 7 4 6 9 1 23

Sample Output 1:

4

Explanation 1: In the given array, the first perfect square to appear is 4.

Sample Input 2:

10 8 14 11 6 15

Sample Output 2:

No

Explanation 2: In the given array, there are no perfect squares.

55. First Element Greater Than K:

You are given an array of integers and another integer K. Print the first element of the array that is greater than K. If there are no elements greater than K, print No.

Sample Input 1:

3 5 10 25 16 12 14
11

Sample Output 1:

25

Explanation 1: In the given array, the first element greater than 11 is 25.

Sample Input 2:

3 5 10 13 16 12 14
19

Sample Output 2:

No

Explanation 2: In the given array, there are no elements greater than 19.

Advanced Array

56. Longest Subarray with Increasing Numbers:

You are given an integer array. Print the length of the longest subarray with increasing numbers.

A subarray is defined as a contiguous portion of an array.

Try not to use nested loop.

Sample Input:

5 4 4 7 6 3 2 4 6 8 6 3 6 8 5

Sample Output:

4

Explanation: The given array has many subarrays with increasing numbers.

4 4 7 -> 5 **4 4 7** 6 3 2 4 6 8 6 3 6 8 5

2 4 6 8 -> 5 4 4 7 6 3 **2 4 6 8** 6 3 6 8 5

3 6 8 -> 5 4 4 7 6 3 2 4 6 8 6 **3 6 8** 5

Out of all these, the subarray with 4 increasing numbers is the longest.

57. Reverse an Array:

You are given an array of integers. Create a new array with elements in reverse order. Print the new array.

Sample Input:

11 1 13 21 3 7

Sample Output:

7 3 21 13 1 11

58. Check Array for Palindrome:

You are given an array of integers. Check if the given array is a Palindrome. If it is a Palindrome array, print yes, else print no.

Note: A Palindrome Array is when the reverse of the array is the same as the original array.

Sample Input 1:

11 1 13 21 3 7

Sample Output 1:

No

Sample Input 2:

17 1 13 1 17

Sample Output 2:

Yes

Explanation 2: The reverse of the array reads same as the original array.

59. Maximum Frequency in an Array:

You are given an array of integers. Find the element that appears the maximum number of times in an array. If multiple elements appear maximum number of times, you can print any.

Sample Input:

5 4 7 11 8 4 6 11 9

Sample Output:

4

Explanation: Both 4 and 11 appear 2 times. We can print any of 4 and 11, so we print 4.

60. Min and Max Difference in an Array:

You are given an array of integers. Find the minimum and maximum difference between any two elements in an array.

Sample Input:

5 4 7 8 4 6 11 9

Sample Output:

0 7

Explanation: Minimum Difference: $4 - 4 = 0$. Maximum Difference: $11 - 4 = 7$

61. Sum of Array Except Self

You are given an array of integers. Print an array where each index has the sum of all numbers in the original array except the number at that index.

Avoid using subtraction, and handle this with nested loops.

Sample Input:

7 3 6 7 5

Sample Output:

21 25 22 21 23

Explanation: The first element of the array is 7, so we print sum of all the elements except the number itself. $21 = 3 + 6 + 7 + 5$.

The second element of the array is 3. $25 = 7 + 6 + 7 + 5$.

The third element of the array is 6. $22 = 7 + 3 + 7 + 5$.

The fourth element of the array is 7. $21 = 7 + 3 + 6 + 5$.

The fifth element of the array is 5. $23 = 7 + 3 + 6 + 7$.

62. Frequency of Each Element in an Array:

You are given an array of integers. Print the frequency of each element in the array.

Sample Input:

3 6 2 1 7 3 7 4 1 7 4

Sample Output:

3 2

6 1

2 1

1 2

7 3

4 2

Explanation: 3 appears twice, but we have to print its frequency only once. Same is with other numbers.

63. Sum of All Differences Between Pairs

You are given an array of integers. Calculate the sum of absolute differences between all pairs of numbers in the array.

Sample Input:

7 3 6 4

Sample Output:

14

Explanation:

Absolute difference between 7 and 3 = 4

Absolute difference between 7 and 6 = 1

Absolute difference between 7 and 4 = 3

Absolute difference between 3 and 6 = 3

Absolute difference between 3 and 4 = 1

Absolute difference between 6 and 4 = 2

Sum of absolute differences between all pairs: $4+1+3+3+1+2 = 14$

64. Find All Pairs with a Given Sum:

You are given an integer array and a target sum. Find all pairs of elements in the array that add up to the given sum.

Sample Input:

4 6 7 2 8 9 3 10

10

Sample Output:

4 6

7 3

2 8

Explanation: The target sum here is 10. $10=4+6$. $10=7+3$. $10=2+8$. Also, if you have printed the pair 4,6 once, you do not need to print it again as 6,4.

65. First Repeat in an Array:

You are given an integer array. Print the first number that repeats itself. If no number repeats in the array, print No.

Sample Input 1:

5 11 4 7 8 4 6 11 7

Sample Output 1:

11

Explanation 1: In the given array, 11 is the first number that repeats itself.

Sample Input 2:

11 1 13 21 3 7

Sample Output 2:

No

Explanation 2: The given array doesn't contain any duplicate element, hence we print No.

66. Subarray with a Given Sum:

You are given an integer array and a target sum. Print a subarray that adds up to the target sum.

If there are multiple possible subarrays, print the one that appears first and is smallest. And if, no such subarray is possible, print "Not Possible".

A subarray is defined as a contiguous portion of an array.

Sample Input 1:

3 6 2 1 7
10

Sample Output 1:

2 1 7

Explanation 1: $10 = 2+1+7$. [2, 1, 7] is a subarray within the given array that adds up to 10.

Sample Input 2:

3 6 2 1 7
14

Sample Output 2:

Not Possible

Explanation 2: No subarray within the given array adds up to 14.

67. Check for a Subarray in an Array

You are given two arrays. Check if the second array is a subarray of the first array. Print yes if it is, else print no.

A subarray is defined as a contiguous portion of an array.

Sample Input 1:

3 6 2 1 7 6 4 9 7
7 6 4 9 7

Sample Output 1:

Yes

Sample Input 2:

3 6 2 1 7 6 4 9 7
7 6 4 9 6

Sample Output 2:

No

68. Common Elements in Two Arrays:

You are given an integer n and two integer arrays each of length n . Print all the common elements between these two arrays.

Print the elements in an order as they appear in the first array. If one common element is repeated multiple times, print it just once. If there are no common elements, print No.

Sample Input 1:

7
4 9 2 5 7 4 3
9 6 4 8 6 1 12

Sample Output 1:

4 9

Explanation 1: In the given arrays, 4 is repeating multiple times, but you need to print it just once. Also, the order of printing 4 and 9 are as per their appearance in the 1st array.

Sample Input 2:

7
2 7 6 4 7 4 3
8 5 9 1 5 8 9

Sample Output 2:

No

Explanation 2: No elements are common in the two given array.

69. Find all Subarrays of an Array:

You are given an integer array. Print all the subarrays.

A subarray is defined as a contiguous portion of an array.

Print the subarrays in an order specified in the sample input / output.

Sample Input:

3 2 1

Sample Output:

3
3 2
3 2 1
2
2 1
1

70. Sum of Subarrays:

You are given an integer array. Print the sum of all possible subarrays.

A subarray is defined as a contiguous portion of an array.

Sample Input:

3 7 5

Sample Output:

52

Explanation: All the subarrays are:

3

3,7

3,7,5

7

7,5

5

Sum = 3 + (3+7) + (3+7+5) + 7 + (7+5) + 5 = 52

Strings

71. Count Vowels and Consonants

You are given a String. Count the number of vowels and consonants the string has. Print the number of vowels followed by the number of consonants.

Note: The string may contain other character like space and full stop.

Vowels are alphabets belonging to the following group - {a, e, i, o u}. Consonants are rest of the alphabets that do not belong to the group - {a, e, i, o u}.

Sample Input:

Hello World

Sample Output:

3 7

Explanation: The string has 3 vowels and 7 consonants.

72. Character Count in a String

You are given a String and a Character. Count how many times the Character is present in the given String. If the Character is not present in the String, print "No".

Sample Input 1:

Hello World

l

Sample Output 1:

3

Explanation 1: The Character l is present 3 times in the String "Hello World".

Sample Input 2:

Hello World

a

Sample Output 2:

No

Explanation 2: The Character a is not present in the String "Hello World".

73. Check for Anagram

You are given two String S1 and S2. Determine if the two strings are anagrams of each other.

Anagrams are words or phrases formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once. For example, the word "listen" is an anagram of "silent".

Note: You will have to ignore case and white spaces.

Sample Input 1:

Silent

Listen

Sample Output 1:

Yes

Explanation 1: Silent and Listen has the same letters. If we ignore case, they are anagrams.

Sample Input 2:

New York Times

monkeys write

Sample Output 2:

Yes

Explanation 2: If we don't consider white space and case-sensitive, "New York Times" and "monkeys write" are anagrams. They have same characters repeating same number of times.

74. Remove Duplicate Characters

You are given a string. Create a new string that contains each character of the original string only once, preserving the order of their first occurrences.

Sample Input:

programming

Sample Output:

progamin

Explanation: We print a new string removing the repeating characters. R, m and g were repeating in the given string "programming" and hence were removed from the new string.

75. Capitalize nth Character

You are given a String and an index. Capitalize the character at the nth position in the given String.

Note: You can consider the index to start from 0.

Sample Input:

programming

6

Sample Output:

prograMming

76. Password Validator

Write a program that takes a string S as input and checks if the string S satisfies all the following conditions to be a strong password:

- Contains at least 8 characters.
- Contains at least one lowercase character.
- Contains at least one uppercase character.
- Contains at least one number.
- Contains at least one special character.

If the string S satisfies all conditions, print yes, else print no.

Sample Input 1:

Abcd@123

Sample Output 1:

Yes

Sample Input 2:

Xyz123

Sample Output 2:

No

77. Search Character in a String:

You are given a string S. You are also given a character c. Check if the character c is present in the string S. If present print yes, else print no.

Sample Input 1:

abcdefgh
f

Sample Output 1:

Yes

Sample Input 2:

abcdefgh
r

Sample Output 2:

No

78. Search Word in a Sentence:

You are given a sentence S. You are also given a word W. Check if the word W is present in the sentence S. If present print yes, else print no.

The input word W will have no space in between.

Sample Input 1:

How are you doing today?
today

Sample Output 1:

Yes

Sample Input 2:

How are you doing today?
do

Sample Output 2:

No