I. Write the following methods that *return a lambda expression* performing a specified action:

1. PerformOperation isOdd(): The lambda expression must return true if a number is odd or  false if it is even.
2. PerformOperation isPrime(): The lambda expression must return true  if a number is prime or false if it is composite.
3. PerformOperation isPalindrome(): The lambda expression must return true  if a number is a palindrome or  false if it is not.

**Input Format**

Input is handled for you by the locked stub code in your editor.

**Output Format**

The locked stub code in your editor will print T lines of output.

**Sample Input**

The first line contains an integer, T (the number of test cases).

The  T subsequent lines each describe a test case in the form of  2 space-separated integers:  
The first integer specifies the condition to check for ( 1 for Odd/Even, 2 for Prime, or 3 for Palindrome). The second integer denotes the number to be checked.

5

1 4

2 5

3 898

1 3

2 12

**Sample Output**

EVEN

PRIME

PALINDROME

ODD

COMPOSITE

II. Write a method that returns 'Fizz' for multiples of three and 'Buzz' for the multiples of five.  
For numbers which are multiples of both three and five return 'FizzBuzz'.  
For numbers that are neither, return the input number.

III. Write a method that returns the nth element of the Fibonacci Sequence  
The Fibonacci Sequence is the series of numbers: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34,...  
The next number is found by adding up the two numbers before it.  
Assume that indexes start at zero, e.g., fib(0) = 0, fib(1) = 1, ...

IV. Write a method that returns the sum of all even Fibonacci numbers. Consider all Fibonacci numbers that are less than or equal to n.  
Each new element in the Fibonacci sequence is generated by adding the previous two elements.  
1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...