# Problem A
# Primitive Roots of a Prime Number

Input File: *testdata.in*
Time Limit: 3 seconds

## Problem Description

An integer p is said to be a prime number if it is greater than 1 and has only factors of 1 and itself. For example, 2, 3, 5, and 97 are prime numbers, but 4, 6, 81, and 95 are not. In modular arithmetic, an integer g, $1 \leq g < p$, is said to be a primitive root modulo p if $\{g, g^2 \bmod p, g^2 \bmod p, \ldots, g^{p-1} \bmod p\}$ = 1, 2, 3, ..., p-1, where (x mod p) means the remainder of x divides p, for example, (13 mod 3) equals 1. This problem asks you to write a program to compute the sum of all primitive roots modulo p for a given prime number p, $3 \leq p \leq 97$.

## Input Format

The input data consists of several lines, each line contains an integer p, $3 \leq p \leq 97$. The end of input is indicated by a number 0. There are at most 5 test cases.

## Output Format

For each case, print the given p, the number of primitive roots of p, and the sum of all primitive roots of p.

## Sample Input

```
3
7
```

```
17
59
89
0
```

## Sample Output

```
3 1 2
7 2 8
17 8 68
59 28 886
89 40 1780
```