(1) (15%) Lucky numbers are defined as follows. Beginning with the natural numbers strike out all even ones, leaving the odd numbers 1, 3, 5, 7, 9, 11, 13, ... The second number is 3, next strike out every third number, leaving 1, 3, 7, 9, 13, ... The third number is 7, next strike out every seventh number and continue this process infinite number of times. The numbers surviving are called lucky numbers. The first few lucky numbers are:

1, 3, 7, 9, 13, 15, 21, 25, 31, 33, … …

Please write a program to generate all lucky numbers between 1 and 1000.

(2) (15%) One of the famous proofs of modern mathematics is Georg Cantor's demonstration that the set of rational numbers is enumerable. The proof works by using an explicit enumeration of rational numbers as shown in the diagram below.

```
1/1  1/2  1/3  1/4  1/5  ...
2/1  2/2  2/3  2/4
3/1  3/2  3/3
4/1  4/2
5/1
```

In the above diagram, the first term is 1/1, the second term is 1/2, the third term is 2/1, the fourth term is 3/1, the fifth term is 2/2, and so on.

Input and Output

Please write a program that will read a list of numbers and will print for each number the corresponding term in Cantor's enumeration as given below. No blank line should appear after the last number. The input list contains a single number per line and will be terminated by end-of-file.

Sample input

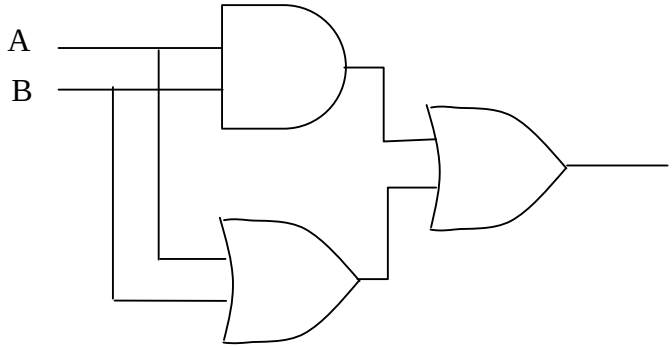
3
14
7

Sample output

Term 3 is 2/1.
Term 14 is 2/4.
Term 7 is 1/4.

(3) (10%) What is a firewall, what does it accomplish, and how does it accomplish it?

(4) (10%) Show the behavior of the following circuit with a truth table.



(5) (10%) Explain what black-box testing and white-box testing are.

(6) (15%) List the output of the following C++ program. This program compares pass-by-value and pass-by-reference with references.

```
#include <iostream>
using namespace std;
int squareByValue(int number)
{
    return number *= number;
}
void squareByReference(int &numberRef)
{
    numberRef *= numberRef
}

int main()
{
    int x = 2;
    int z = 4;
```

```
    cout << "x= " << x << " before squareByValue\n";
    cout << "value Returned by squareByValue: " << squareByValue(x) << endl;
    cout << "x= " << x << " after squareByValue\n";

    cout << "z= " << z << " before squareByReference\n";
    squareByReference(z);
    cout << "z= " << z << " after squareByReference\n";
}
```

(7) (10%) What does a compiler do? What does an interpreter do?

(8) (10%) One commonly used software design pattern is to divide the application software into layers. For example, software can be divided into view layer, business logic layer, and data layer. The view layer accepts user input and display processing results. The business logic layer implements the rules and procedures of business processing. The data layer manages stored data, usually in one or more databases. Explain the advantages of this layered architecture.

(9) (5%) The ASCII code of letter 'A' is 41 in hexadecimal format. Find the ASCII code of letter 'Z' in decimal format.