

A

A Giveaway

Input: Standard Input
 Output: Standard Output

A positive integer number is "Special" if it is both a square (eg. **1, 4, 9, 16, 64 ...**) and a cube (eg. **1, 8, 27, 64 ...**). The smallest special number is **1**. Now your job is to write a program that finds whether a number less than **100000000** is special or not. It may be noted that there are only **21** such numbers within this range and these are **1, 64, 729, 4096, 15625, 46656, 117649, 262144, 531441, 1000000, 1771561, 2985984, 4826809, 7529536, 11390625, 16777216, 24137569, 34012224, 47045881, 64000000** and **85766121**. A very childish but legitimate C/C++ solution, which would work for positive numbers not exceeding **15624**, is shown below.

```
#include<stdio.h>
int main(void)
{
    int num;
    while (scanf("%d",&num) && num)
    {
        if(num==1 || num==64 || num==729 || num==4096)
            printf("Special\n");
        else
            printf("Ordinary\n");
    }
    return 0;
}
```

A C/C++ code that will work for positive numbers not exceeding 15624

Input

The input file contains at most **1001** lines of input. Each line contains a positive integer less than **100000000**. Input is terminated by a line containing a zero.

Output

For each line of input except the last one produce one line of output. This line contains a string (without the quotes) "**Special**" if the number is special and "**Ordinary**" if the number is not special. Look at the output for the sample input for details.

Sample Input

Output for Sample Input

| | |
|-------|----------|
| 1 | Special |
| 2 | Ordinary |
| 64 | Special |
| 100 | Ordinary |
| 15625 | Special |
| 0 | |