**Bài: Viên ngọc kiên cường**

Trong vương quốc Numerica huyền bí, viên ngọc Sức mạnh Vĩnh cửu từ lâu đã là biểu tượng bảo vệ cho sự thịnh vượng và bình an. Tuy nhiên, ác quỷ Digiton đã giáng lời nguyền tàn khốc lên nó, khiến sức mạnh của viên ngọc bị suy yếu sau mỗi đòn tấn công. Cụ thể, sức mạnh của viên ngọc, vốn được biểu diễn bằng một con số, sẽ trải qua phép biến đổi: lấy tích các chữ số hiện tại để tạo thành sức mạnh mới. Quá trình này lặp lại cho đến khi sức mạnh chỉ còn là một chữ số duy nhất, và khi đó viên ngọc sẽ tan vỡ, mất đi ánh sáng bảo vệ vương quốc.

Trước lời nguyền hiểm ác này, nhà vua Numerica ra lệnh tìm cách xác định khả năng chống chịu của viên ngọc bằng cách tính số lần biến đổi cần thiết để viên ngọc tan vỡ. Đồng thời, để hiểu rõ hơn về sức mạnh của các viên ngọc khác trong vương quốc, nhà vua yêu cầu tính xem số lần chống chịu của viên ngọc có sức mạnh **n** và viên ngọc chống chịu được nhiều lần tấn công nhất trước khi bị phá hủy trong các viên ngọc có sức mạnh từ 1 đến **n**,. Nếu có nhiều viên ngọc cùng số lần chống chịu tối đa, nhà vua sẽ chọn viên ngọc có sức mạnh ban đầu lớn nhất.

Chẳng hạn, với n = 100, viên ngọc có sức mạnh ban đầu 77 sẽ được vinh danh là viên ngọc mạnh nhất khi chống chịu được 4 lần tấn công của Digiton, nhiều hơn bất kỳ viên ngọc nào khác trong cùng khoảng sức mạnh.

**Dữ liệu vào:**

* Đầu vào vào một số nguyên dương **n, ()**

**Dữ liệu ra:**

* Dòng đầu tiên chính là sức chống chịu của viên ngọc có sức mạnh n
* Dòng thứ hai gồm 2 số đó là chính là sức mạnh và độ chống chịu của viên ngọc có độ chống chịu lớn nhất, nếu có nhiều viên ngọc cùng độ chống chịu lớn nhất chọn viên ngọc có sức mạnh lớn nhất

**Ràng buộc:**

* 70% số điểm với  **()**
* 30% số điểm với dữ liệu đúng với dữ liệu vào

**Ví dụ:**

|  |  |
| --- | --- |
| **RESILIENTGEM.INP** | **RESILIENTGEM.OUT** |
| 33 | 2  29 |

Viên ngọc có sức mạnh 33 chống chịu lần 1 số còn sức mạnh là 9, sau lần thứ 2 thì viên ngọc bị vỡ

Từ 1 đến 33 thì viên ngọc 29 là viên ngọc chống chịu được nhiều lần nhất

Hướng dẫn giải

#include<bits/stdc++.h>

using *namespace* std;

*int* mulDigits(*int* *n*) {

*int* res = 1;

    while (n > 0) {

        res \*= n % 10;

        n /= 10;

    }

    return res;

}

*int* main()

{

    ifstream cin("RESILIENTGEM.INP");

    ofstream cout("RESILIENTGEM.OUT");

*int* n;

    cin >> n;

    vector<*int*> dp(n + 1, 1);

    for (*int* i = 10; i <= n; i++) {

        dp[i] = dp[mulDigits(i)] + 1;

    }

*int* res = 1, maxVal = 1;

    for (*int* i = 1; i <= n; i++) {

        if (dp[i] >= maxVal) {

            maxVal = dp[i];

            res = i;

        }

    }

    cout << dp[n] << endl;

    cout << res << " " << maxVal << endl;

}

**1. Ý tưởng chính**:

* Đầu vào là một số nnn. Chúng ta cần tìm số có khả năng "chống chịu" nhiều nhất từ 1 đến nnn, tức là số biến đổi cần thiết để giảm sức mạnh của số đó (biến đổi thành một chữ số duy nhất).

**2. Chống chịu của viên ngọc**:

* Mỗi số có thể trải qua nhiều lần biến đổi bằng cách lấy tích các chữ số của nó. Ví dụ, số 39 sẽ có: 3×9=27 rồi tiếp tục 2×7=14, và cứ thế cho đến khi đạt được một số đơn lẻ.

**3. Cách tiếp cận:**

* Dùng một mảng dp để lưu trữ số lần biến đổi (chống chịu) của mỗi số từ 1 đến nnn. Ban đầu, số 1 sẽ có số lần biến đổi là 1 vì nó đã là một số đơn lẻ.
* Với mỗi số iii từ 10 trở đi, tính số lần biến đổi bằng cách lấy giá trị của dp[mulDigits(i)]+1dp[mulDigits(i)] + 1dp[mulDigits(i)]+1, trong đó mulDigits(i) là hàm tính tích các chữ số của iii.

**4. Tìm số viên ngọc mạnh nhất**:

* Duyệt qua mảng dp để tìm số có khả năng chống chịu nhiều nhất, tức là có số lần biến đổi lớn nhất. Nếu có nhiều số có số lần biến đổi giống nhau, chọn số có giá trị lớn nhất.

**5. Kết quả:**

* In ra số lần biến đổi của viên ngọc có sức mạnh là nnn.
* In ra số có khả năng chống chịu nhiều nhất và số lần chống chịu của nó.