

Temas:

Grafos explícitos:

Grafo transversal

- DFS (Mario)
- BFS (Mario)
- Topological sort (Mario)
- Bipartite Graph Check (Mario)
- Propiedad de chequeo de aristas de los grafos mediante el DFS árbol de expansión (Juan)
- Encontrar puntos de articulación y puentes (Juan)
- Encontrando los componentes fuertemente conectados (Mario)

Árbol de expansión mínima

- Algoritmo de Prim (Juan)
- Kruskal (Juan)
- Conjuntos disjuntos (Juan)

Camino más corto desde un origen único.

- SSSP en grafos sin pesos (Juan)
- SSSP en grafos con pesos (Mario)
 1. Dijkstra (Mario)
 2. Bellman Ford (Mario)
 3. Floyd Warshall(juan)

Grafos implícitos:

- FloodFill (Mario)

Todo lo de Shifting: (Juan)

Manipulación de bits

- Representación y Manipulación de Bits
- Bit Shifts
- BitMasks
- Sets y Creación de Sub-sets + Operaciones para Sets
- Hamming
- Contar Subgrids

Búsqueda Completa

- Generación de Sub-sets

RPC PASADAS:

Problem G Organ Freeman solved by: Juan



Competitive Programming Network - 1st Activity

February 25th, 2023

Problem G. Organ-free Man

Source file name: Organ.c, Organ.cpp, Organ.java, Organ.py

Input: Standard

Output: Standard

Every so often, a shipment of universal robots comes from Earth to Mars in order to help you with routine colonization tasks. The robots are called Organ-free Men (precisely OFMv5001.41.912) and each one of them is identified by a unique serial number, which is a positive integer.

To prevent space theft, OFMs are transported from Earth to Mars in a locked state and have to be first unlocked by a special password. The password to unlock an OFM is based on its serial number and a function $f(x)$. The function $f(x)$ is defined as follows:

If $0 \leq x \leq 9$, then $f(x) = x!$, and if $x > 9$, then $f(x) = (x \bmod 10)! + f(\lfloor x/10 \rfloor)$. The brackets $\lfloor \cdot \rfloor$ denote the floor value of a number (e.g. $\lfloor 2.43 \rfloor = 2$). Exclamation mark denotes the factorial, i.e., $x! = 1 \cdot 2 \cdot \dots \cdot x$ for $x > 0$ and $0! = 1$.

To unlock an OFM with a serial number y , you need to input smallest such non-negative integer x , so that $f(x) = y$ holds.

Will you manage to unlock all robots that were shipped to you?

Input

The input consists of one integer y ($1 \leq y \leq 10^9$), the serial number of a particular OFM.

Output

Output a single non-negative integer x , the password to unlock the particular OFM.

Example

Input	Output
3	12
20	2333

Manipulación de bits

Problem K KIARA is a Recursive Acronym solved by: Juan

ICPC Latin American Regional – 2021

Problem K – KIARA is a Recursive Acronym

A recursive acronym is an acronym in which one of its letters stands for the acronym itself. For instance, the first word in the title of this problem is a recursive acronym of the full title. Another example is “BOB”, which is an acronym of “Beware of Bob”.

Given a list of words, you must decide whether there exists a word in the list which is a recursive acronym of a phrase that can be formed using words in the list. Since the first letter of any word can stand for the whole word, it is enough to decide whether there exists a word in the list which can be formed using the first letter of some words in the list.

Input

The first line contains a positive integer N indicating the number of words in the list. Each of the next N lines contains a non-empty string made of uppercase letters representing a word in the list. The sum of the lengths of all the strings is at most 10^6 .

Output

Output a single line with the uppercase letter “Y” if there exists a word in the list which is a recursive acronym of a phrase that can be formed using words in the list, and the uppercase letter “N” otherwise.

Sample input 1 3 OF BOB BEWARE	Sample output 1 Y
Sample input 2 3 WHO MADE WHO	Sample output 2 N
Sample input 3 5 JUST USE WORD XX TWICE	Sample output 3 Y
Sample input 4 1 YYYYYYYYYY	Sample output 4 Y

Manipulación de bits

SPOJ: Sphere Online Judge

Toposort - Topological Sorting solved by: **Mario, Laura, Juan**

<https://www.spoj.com/problems/TOPOSORT/> topological sort

CSES:

Course Schedule solved by: **Mario, Juan**

<https://cses.fi/problemset/task/1679> topological sort

Weird Algorithm solved by: **Juan**

<https://cses.fi/problemset/task/1068> Operaciones Bitwise

Beecrowds,

1026 - To Carry or not to Carry solved by: **Juan**

<https://www.beecrowd.com.br/judge/en/problems/view/1026> Operaciones Bitwise

1152 solved by: **Juan, Jhosua, Laura, Mario**

<https://www.beecrowd.com.br/judge/en/problems/view/1152> Kruskal disjunto o MST

1148 solved by: **Juan, Mario**

<https://www.beecrowd.com.br/judge/en/problems/view/1148> Dijkstra

1076 solved by: **Mario, Juan, Laura**

<https://www.beecrowd.com.br/judge/es/problems/view/1076> Dfs

3171 solved by: **Mario, Juan, Laura**

<https://www.beecrowd.com.br/judge/es/problems/view/3171> Dfs o Dfs o Disjunto

1128 solved by: **Mario, Juan, Laura**

<https://www.beecrowd.com.br/judge/es/problems/view/1128> Dfs o Bfs

1082 solved by: **Mario, Juan**

<https://www.beecrowd.com.br/judge/es/problems/view/1082> Dfs o Bfs

1583 solved by: **Mario, Juan, Laura**

<https://www.beecrowd.com.br/judge/es/problems/view/1583> Floodfill

1907 solved by: **Juan, Jhosua**

<https://www.beecrowd.com.br/judge/en/problems/view/1907> Floodfill en Bfs

UVA ONLINE JUDGE

11054 - Wine trading in Gergovia solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=22&page=solve_problem&problem=1995 Bitmask y Operaciones Bitwise

10982 - Troublemakers solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=21&page=show_problem&problem=1923 **Bitmask o Bipartite Grafo**

729 - The Hamming Distance Problem solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=9&page=show_problem&problem=670 **Hamming**

12455 - Bars solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=279&page=show_problem&problem=3886 **Sub-sets**

725 - Division solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=9&page=show_problem&problem=666 **Busqueda Completa**

700 - Date Bugs solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=9&page=show_problem&problem=641 **Busqueda Completa**

11173 - Grey Codes solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=23&page=show_problem&problem=2114 **Manipulación de Bits**

594 - One Little, Two Little, Three Little Endians solved by:

https://onlinejudge.org/index.php?option=onlinejudge&Itemid=8&page=show_problem&problem=535 **Manipulación de Bits**

11933 - Splitting Numbers solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=229&page=show_problem&problem=3084 **Manipulación de Bits y Bitwise**

11172 - Relational Operator solved by: Juan

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=23&page=show_problem&problem=2113 **Manipulación de Bits y Bitwise**

469 - Wetlands of Florida solved by: Mario, Juan

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=667&page=show_problem&problem=410 **floodfill**

572 - Oil Deposits solved by: Mario, Juan, Laura

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=667&page=show_problem&problem=513 **floodfill**

10336 - Rank the Languages solved by: Mario, Juan, Laura

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=667&page=show_problem&problem=1277 floodfill

11244 - Counting Stars solved by: Mario, Juan, Laura

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=667&page=show_problem&problem=2201 floodfill

908 - Reconnecting computer Sites solved by: Juan, Jhosua, Mario

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=673&page=show_problem&problem=849 Kruskal disjuncto

1208 - Oreon solved by: Juan, Jhosua

https://onlinejudge.org/index.php?option=onlinejudge&Itemid=8&page=show_problem&problem=3649 Kruskal

11228 - Transportation System solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=673&page=show_problem&problem=2169

11631 - Dark roads solved by: Juan, Jhosua, Mario

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=673&page=show_problem&problem=2678 MST

2847- Heavy Cycle Edges solved by:

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=673&page=show_problem&problem=2847

10305 - Ordering tasks solved by: Mario, Juan

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=668&page=show_problem&problem=1246 topological sort Nota: El resultado real del caso de prueba es: 4 1 5 2 3

200 - Rare Order solved by: Mario, Juan

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=668&page=show_problem&problem=136 topological sort

929 - Number Maze solved by: Mario, Juan

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=678&page=show_problem&problem=870 Dijkstra

1112 - Mice and Maze solved by: Mario

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=678&page=show_problem&problem=3553 Dijkstra

10986 - Sending email solved by: Mario

https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=678&page=show_problem&problem=1927 Dijkstra

nooooo00000o0 jjaja mas ejercicios jajaj mas ejercicios :c

Definición de 1 un grafo usando tuplas: Se utiliza cuando se requiere información adicional de las aristas

```
typedef tuple < int, int, int > ti;  
typedef vector < ti > edgeList;
```

```
int main ( ) {  
    int n, e;  
    cin >> n >> e;  
    while (e-->0) {  
        int a, b, w;  
        cin >> a >> b >> w;  
        edgeList.push_back ( make_tuple ( a, b, w ) );  
    }  
    for (const auto& edge : edgeList) {  
        int a, b, w;  
        tie(a, b, w) = edge; // El tie se usa para desempaquetar los elementos de la tupla  
        // Para poder imprimirlos  
        cout << a << " " << b << " " << w << "\n";  
    }  
    return 0;  
}
```