

Problema A

Lluvia

Nombre de la fuente del archivo: lluvia.c, lluvia.cpp o lluvia.java

Usted debe leer de la entrada estándar y escribir en la salida estándar

Un jueves por la tarde, las nubes escondían al sol, mientras la dulce lluvia mojaba suavemente la hierba y nos dejaba una delicada sensación de paz y soledad. Las horas parecen más largas, que no se terminan nunca, alguien podría decir, mientras la gente mira el agua azul y fría que cae del cielo, que sus mentes se dejan llevar.

Pasar el tiempo alrededor de este día, Juan y John juegan un juego nuevo. Tienen un montón de piedras, Juan juega primero, después John, y luego Juan de nuevo, y así sucesivamente, hasta que el juego termina. El juego continúa de esta manera, en cada turno el jugador debe tomar una de las piedras y retirar de la pila, al lado del otro jugador toma su turno. Si no hay una piedra a tomar, entonces el jugador pierde.

Teniendo en cuenta que usted es consciente de la cantidad de piedras, ¿quién ganaría?

Entrada:

Hay varios casos de prueba, cada uno consiste en una línea con un entero N, el número de piedras de la pila del juego. La entrada termina con N igual a 0, este caso no debería ser procesado.

Salida:

Para cada caso, imprima en una línea el nombre del jugador que gana el juego.

Ejemplo de Entrada

Ejemplo de Salida

3	Juan
8	John
0	

Problema B

Dementes

Nombre de la fuente del archivo: dementes.c, dementes.cpp o dementes.java

Usted debe leer de la entrada estándar y escribir en la salida estándar

Sesom y Susej son dos tipos muy dementes, pero a su vez son muy amigos. Su amistad es muy extraña, a tal punto que Susej pareciera amar a Sesom. Un día salieron a dar un paseo pero como de costumbre terminaron discutiendo, llegando al extremo de comenzar a pelear. No sabemos exactamente que pasó en dicha pelea, porque una versión dice que fue toda una batalla mientras que Sesom no recuerda absolutamente nada, suponemos que el golpe que le propinaron en la nariz lo hizo perder todos sus recuerdos.

Hoy en día se encuentran juntos de nuevo a pesar de haber peleado (recuerden que son dementes) y están haciendo una pequeña tarea. Susej ya terminó su parte pero Sesom todavía debe realizar un programa que dada una palabra verifique si es palíndromo (una palabra es palíndromo si puede leerse igual de derecha a izquierda y de izquierda a derecha). Sesom no es un buen programador, así que por favor ayúdalo con esta tarea o sino Susej le romperá la nariz otra vez.

Entrada:

La entrada comienza con un entero N que representa la cantidad de palabras a leer. Seguido se encuentran N palabras todas conformadas únicamente de letras en minúsculas.

Salida:

Por cada palabra imprimir una línea, esta debe ser "SI" si la palabra es palíndromo o "NO" en cualquier otro caso.

Ejemplo de Entrada

Ejemplo de Salida

2 sesom arepera	NO SI
-----------------------	----------

Problema C

Hola Mundo

Nombre de la fuente del archivo: *holamundo.c, holamundo.cpp o holamundo.java*

Usted debe leer de la entrada estándar y escribir en la salida estándar

El hola mundo, ejemplo básico para programadores y no programadores alrededor del planeta. En búsqueda de nuevos retos se ha de ofrecer un nuevo problema que consiste en imprimir no solo un Hola Mundo sino más bien muchos Hola Mundo.

Entrada

La entrada consiste en varios casos de prueba, cada uno contiene un entero N. El programa debe terminar al leer un 0, este caso no debe ser procesado.

Salida

Por cada caso de prueba imprimir N líneas que contengan "Hola Mundo", sin comillas. Imprimir una línea en blanco luego de cada caso.

Ejemplo de Entrada

Ejemplo de Salida

3	Hola Mundo
2	Hola Mundo
0	Hola Mundo
	Hola Mundo
	Hola Mundo

Problema D

Pingüinos

Nombre de la fuente del archivo: ping.c, ping.cpp o ping.java

Usted debe leer de la entrada estándar y escribir en la salida estándar

El señor sin cuello, charlatán como pocos, se ha encontrado un difícil problema de algoritmia, o así dice él. Resulta que, encantado con pingüinos del todo el mundo, ha considerado a uno de ellos como el más importante, sin embargo no recuerda cual es.

Entonces ha decidido anotar la altura de cada pingüino y va a considerar a aquel más alto como el nuevo más importante. Pero no sabe cómo resolver el problema y te ha pedido tu ayuda.

Entrada

La entrada comienza con un entero T, el cual es la cantidad de casos de prueba. Cada caso de prueba consiste en dos líneas la primera contiene un entero C , la cantidad de pingüinos que se midieron, y la segunda tiene números enteros correspondientes a los tamaños de los pingüinos. En un mismo caso de prueba todos los pingüinos tendrán estaturas distintas.

Salida

Por cada caso imprima el lugar que ocupa el pingüino más alto.

Ejemplo de Entrada

Ejemplo de Salida

3	2
2	5
18 45	4
8	
2 3 58 98 99 6 8 76	
5	
3 4 2 63 3	

Problema E

Primos

Nombre de la fuente del archivo: *primos.c, primos.cpp o primos.java*

Usted debe leer de la entrada estándar y escribir en la salida estándar

Droopy es un joven que le gusta mucho la computación, en especial las competencias de programación. Un día estaba estudiando la representación de los números primos y quedó sorprendido al ver que la cantidad de ellos son menores a la totalidad de todos los números. Luego de estudiar por un par de horas concluyó que los números primos son más atractivos que los demás.

Ahora se le solicita que elabore un programa que dado un número en su representación diga si es primo o no.

Entrada

Existe varios casos, cada uno comienza con un entero N el número de casos de prueba existentes, seguidos de N líneas que contienen un número entero cada una.

Salida

Por cada caso imprima una línea donde diga "PRIMO" si el número es primo o "NO PRIMO" si no lo es.

Ejemplo de Entrada

Ejemplo de Salida

4	PRIMO
2	PRIMO
3	NO PRIMO
22	PRIMO
97	

Problema F

Codificación

Nombre de la fuente del archivo: codif.c, codif.cpp o codif.java

Usted debe leer de la entrada estándar y escribir en la salida estándar

En muchas competencias de programación donde se participa es necesario convertir y transformar códigos indescifrables para los competidores, es un trabajo duro y que pocos son capaces de lograr pero Peter famoso por resolver los códigos de comunicación entre las divisiones de guerra y militares en todo el mundo se colocó la tarea de explicarles a un grupo de competidores como lo logra, les dice que los mensajes ocultos que pueden estar escritos son una tarea complicada pero cuando lo logra siente una alegría incomparable.

El trabajo que tiene Peter es descifrar este mensaje para poder demostrarle al grupo todos los reconocimientos que ha tenido en esta difícil tarea.

Entrada

Existe varios casos, cada uno comienza con un entero C el número de casos de prueba existentes, seguido de C líneas con una palabra cada una para ser descifradas. Cada palabra está formada únicamente de letras en minúscula.

Salida

Por cada caso imprima una línea donde muestre la solución que encontró Peter.

Ejemplo de Entrada

Ejemplo de Salida

6 bdn nbsbupoft ovfwpt qsphsbnbepsft mjtupt abqbsp	acm maratones nuevos programadores listos zapato
--	---

Problema G

Velocidad

Nombre de la fuente del archivo: *velocidad.c*, *velocidad.cpp* o *velocidad.java*

Usted debe leer de la entrada estándar y escribir en la salida estándar

Una partícula viaja a una velocidad dada y tiene una aceleración constante. Se observa su trayectoria durante un tiempo t , medido en minutos. Si se conoce el tiempo que ha estado en movimiento la partícula y la velocidad con la que esta se desplaza, ¿es posible saber qué distancia habrá recorrido la partícula en el doble del tiempo registrado?

Entrada

Cada caso de prueba consistirá en dos enteros en una línea. Estos enteros son V y T , V será la velocidad de la partícula ($100 \leq V \leq 1000$) y T el tiempo en que ha estado en movimiento la partícula ($0 \leq T \leq 200$). La entrada termina cuando V y T son iguales a cero.

Salida

Para cada línea de entrada debe imprimir un solo número entero en una línea que indica el desplazamiento en el doble de ese tiempo.

Ejemplo de Entrada

Ejemplo de Salida

1 2	4
5 6	60
0 0	

Problema H

Chiquipizzas

Nombre de la fuente del archivo: *pizzas.c*, *pizzas.cpp* o *pizzas.java*

Usted debe leer de la entrada estándar y escribe en la salida estándar

Hay un problema en un restaurante italiano cerca de mi casa, tienen un problema con la máquina de corte, que quieren reducir sus pedazos pequeños de pizzas, por lo que se cortó rebanadas a través del diámetro de la pizza, pero no saben cuántas piezas van a conseguir al cortar.

Sabiendo el numero de cortes que se le harán a la pizza ¿Puede predecir la cantidad de rebanadas que se obtendrán?

Entrada

Hay varios casos de prueba en la entrada, cada uno con el valor de n , un número entero entre 1 y 100 inclusive. El último caso es seguido por $n = 0$, este caso no se debe procesar.

Salida

Para cada caso de uso se debe indicar los pedazos de pizza que se obtendrán en el formato que se le indica.

Ejemplo de Entrada

Ejemplo de Salida

3	Caso 1: 6
5	Caso 2: 10
10	Caso 3: 20
0	

Soluciones

Problema A

Lluvia

El juego de Juan y John consiste simplemente en remover una a una las piedras de un montículo. Como Juan siempre es el primero en agarrar una piedra, podemos deducir que siempre que el número de piedras sea impar Juan ganará y cada vez que sean pares John será el ganador.

```
#include <iostream>
using namespace std;
int main(){
    int n;
    while((cin >> n) && n != 0){
        if(n % 2 != 0)
            cout << "Juan" << endl;
        else
            cout << "John" << endl;
    }
    return 0;
}
```

Problema B

Dementes

En este ejercicio el objetivo es verificar si una palabra es palíndromo, es decir que se lee igual de atrás hacia adelante. Para verificar esto usaremos un ciclo for para recorrer la palabra leída, pero en este caso la recorreremos usando dos índices: *i* para recorrer de principio hasta la mitad, y *j* para recorrerla desde el final hasta la mitad. Luego por cada iteración se verifica si las dos letras a las que accedemos con *i* y *j* son iguales, si no lo son ya sabemos que la palabra no puede ser palíndromo y por lo tanto nos salimos del ciclo.

```
#include <iostream>
using namespace std;
int main(){
    string palabra;
    int cantidad;
    bool esPalindromo;
    cin >> cantidad;
    while(cantidad--){
        esPalindromo = true;
        cin >> palabra;
        for(int i = 0, j = palabra.size()-1; i < j; i++, j--){
            if(palabra[i] != palabra[j]){
                esPalindromo = false;
                break;
            }
        }
    }
}
```

```

        if(esPalindromo)
            cout << "SI" << endl;
        else
            cout << "NO" << endl;
    }
    return 0;
}

```

Problema C
Hola Mundo

```

import java.util.Scanner;
class holamundo{
    public static void main(String[] args){
        int cantidad;
        Scanner lector = new Scanner(System.in);
        while((cantidad = lector.nextInt()) != 0){
            for(int i = 0; i < cantidad; i++){
                System.out.println("Hola Mundo");
                System.out.println("");
            }
        }
    }
}

```

Problema D
Pingüinos

El fin de este ejercicio es simplemente mostrar la posición del pingüino mas alto.

```

#include <iostream>

using namespace std;

int main(){
    int casos, cantidad, mayor, actual, lugar;
    cin >> casos;
    while(casos--){
        cin >> cantidad;
        mayor = 0;
        for(int i = 0; i < cantidad; i++){
            cin >> actual;
            if(actual > mayor){
                mayor = actual;
                lugar = i + 1;
            }
        }
        cout << lugar << endl;
    }
    return 0;
}

```

Problema E
Primos

El ejercicio se basa en leer una serie de números y verificar si esos números son primos o no. Una manera de resolver el ejercicio es utilizar la Criba de Eratóstenes, con la cual, podemos conocer todos los números primos que se encuentran entre dos y un número N.

A continuación la solución con la criba de Eratóstenes.

```
#include <iostream>
#include <cstring>

using namespace std;

int main(){
    const int tam = 101;
    bool primos[tam];
    primos[2] = true;
    for(int i = 0; i < tam; i++) primos[i] = true;
    for(int i = 2; i*i <= tam; i++){
        if(primos[i])
            for(int j = 2; i*j < tam; j++){
                primos[j*i] = false;
            }
    }

    int casos, nro;
    cin >> casos;
    while(casos--){
        cin >> nro;
        if(primos[nro])
            cout << "PRIMO" << endl;
        else
            cout << "NO PRIMO" << endl;
    }
    return 0;
}
```

Problema F
Codificación

En este ejercicio la codificación utilizada consiste en reemplazar cada letra del alfabeto por la que le sigue, por lo tanto la "a" se reemplaza por la "b", la "b" por la "c" y así sucesivamente, en el caso de la "z" esta es reemplazada por "a". Por lo tanto si queremos codificar las palabras que nos proporcionan es necesario cambiar cada letra por la siguiente.

```
#include <iostream>

using namespace std;

int main(){
    string palabra, solucion;
    int cantidad;
    cin >> cantidad;
    while(cantidad--){
        cin >> palabra;
        solucion = "";
        for(int i = 0; i < palabra.size(); i++){
            if(palabra[i] != 'a')
                solucion += palabra[i]-1;
            else
                solucion += 'z';
        }
        cout << solucion << endl;
    }
    return 0;
}
```

Problema G
Velocidad

```
#include <iostream>

using namespace std;

int main(){
    int V,T;
    while((cin >> V >> T) && V != 0 && T != 0){
        cout << V*T*2 << endl;
    }
    return 0;
}
```

Problema H
Chiquipizzas

```
#include <iostream>

using namespace std;

int main(){
    int cortes, contador = 1;
    while((cin >> cortes) && cortes != 0){
        cout << "Caso " << contador << ": " << cortes * 2 << endl;
        contador ++;
    }
    return 0;
}
```