



ACM-UCLA Programación Creativa 2012

Práctica Dos

19 de Mayo de 2012

Universidad Centroccidental Lisandro Alvarado

Este problemario contiene 9 problemas; las páginas están enumeradas de 1 a 9.



Problema A

Ámsterdam

Nombre de archivo: amsterdam.c, amsterdam.cpp o amsterdam.java
Debe leer desde entrada estándar e imprimir a salida estándar.

Se conoce que la ciudad *Ámsterdam* es visitada constantemente por turistas de todos los lugares del mundo, los cuales pueden hablar distintos idiomas. Un día un tipo sin oficio enumeró a los N turistas en el rango $0..N-1$ y comenzó a tomar nota de qué turistas entablaban una conversación a lo largo del día con la intención de saber cuántos idiomas distintos pueden hablar los N turistas. Al final del día, tiene toda su información en una pequeña libreta, pero está triste porque a pesar de su esfuerzo no puede conocer ese número esperado, así que le pide a usted que le diga cuál es la mayor cantidad de idiomas que hablan los turistas utilizando la información que se encuentra en la libreta. Se asume que cada turista habla un único idioma.

Entrada

La primera línea señala el número de casos de prueba T ($T \leq 10$). Cada caso de prueba comienza con 2 enteros N ($2 \leq N \leq 100$) y K ($0 \leq K \leq 150$) en una línea, que indican los N turistas y las K conversaciones que fueron observadas. Luego siguen K líneas con 2 enteros cada uno que son los índices de los turistas que fueron observados mientras conversaban.

Salida

Por cada caso imprimir una línea con un simple entero indicando el máximo de idiomas que pueden ser hablados por los turistas.

Ejemplo de entrada

```
1 1
2 4 2
3 0 2
4 1 3
5
```

Ejemplo de Salida

```
1 2
2
```



Problema B

(Problema de) Bits II

Nombre de archivo: mbits.c, mbits.cpp o mbits.java
Debe leer desde entrada estándar e imprimir a salida estándar.

Cada entero puede ser representado en un sistema binario, tenemos a 5 en base 10 corresponde a 101 en base 2. Y podemos definir como cantidad de bits activos al total de dígitos 1 que aparecen en la representación binaria de un número en particular, en este ejemplo 5 tiene dos bits activos.

El problema consiste en, dado dos enteros A y B , determinar el número X que se encuentre en el rango entre A y B , inclusive, con mayor cantidad de bits activos. De haber un empate entre dos o más números, buscar el menor de ellos.

Entrada

La entrada comienza con un entero T ($T \leq 10^4$) que indica el número de casos. Siguen T líneas, cada uno con un par de enteros A y B ($0 \leq A \leq B \leq 10^6$).

Salida

Por cada caso imprimir una línea con el valor de X .

Ejemplo de entrada

```
1 6
2 0 10
3 3 4
4 5 6
5 0 0
6 1 100
7 0 1000000
8
```

Ejemplo de Salida

```
1 7
2 3
3 5
4 0
5 63
6 524287
7
```



Problema C

Contando Primos

Nombre de archivo: cp.c, cp.cpp o cp.java
Debe leer desde entrada estándar e imprimir a salida estándar.

Un número entero puede ser representado como la suma de uno o más números primos, por ejemplo el 8 puede ser expresado como la suma de 3 y 5.

Dado un número entero N , buscar cuantas combinaciones diferentes de K números primos distintos se puede construir con tal de que la suma de estos dé exactamente N . Por ejemplo con $N = 24$ y $K = 2$ la respuesta es tres ya que podemos construir 3 combinaciones: $5 + 19$, $7 + 17$ y $11 + 13$. En este problema el 1 no se considera como un número primo.

Entrada

Un entero $T \leq 20000$ que indica el número de casos, seguido de T líneas con los enteros N y K , donde $N \leq 1120$ y $K \leq 14$.

Salida

Por cada caso imprimirás una línea con el número de formas.

Ejemplo de entrada

```
1 5
2 24 2
3 100 5
4 1000 10
5 1120 13
6 1120 14
7
```

Ejemplo de Salida

```
1 3
2 55
3 200102899
4 1581918613
5 2079324314
```



Problema D

Dementes

Nombre de archivo: dementes.c, dementes.cpp o dementes.java

Debe leer desde entrada estándar e imprimir a 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 palíndrome (una palabra es palíndrome si puede leerse igual de derecha a izquierda y de izquierda a derecha). Además, si asumiendo que todas las letras son mayúsculas obtenemos un palíndrome diremos que la palabra original es “casi” palíndrome. Sesom no es un buen programador, así que por favor ayúdelo con esta tarea o sino Susej le romperá la nariz otra vez.

Entrada

Una palabra por línea, la cual estará formada por los caracteres a...z y A...Z con una longitud máxima de 20. La entrada finaliza con EOF.

Salida

Por cada caso imprimir una línea, esta debe ser “SI” si la palabra es palíndrome, “CASI” si se puede transformar en palíndrome o “NO” en cualquier otro caso.

Ejemplo de entrada

```
1 Sesom
2 arepera
3 ArepeRa
4
```

Ejemplo de Salida

```
1 NO
2 SI
3 CASI
4
```



Problema E

¡Estos Divisores!

Nombre de archivo: ed.c, ed.cpp o ed.java

Debe leer desde entrada estándar e imprimir a salida estándar.

Este es un problema muy simple, tu tarea es buscar cual es el número más pequeño que puede ser dividido (división exacta) por un rango de números $[A, B]$. Por ejemplo el 2520 es el número más pequeño el cual puede ser dividido exactamente por todos los números en el rango $[1, 10]$.

Entrada

Un número N el cual indica cuantos casos se van a leer, por cada N líneas se encontrarán dos enteros A y B donde $1 \leq A \leq B \leq 40$.

Salida

Una línea imprimiendo el número requerido.

Ejemplo de entrada

```
1 2
2 1 10
3 20 40
4
```

Ejemplo de Salida

```
1 2520
2 5342931457063200
3
```



Problema F

Flash

Nombre de archivo: flash.c, flash.cpp o flash.java
Debe leer desde entrada estándar e imprimir a salida estándar.

Flash es un joven superdotado con cualidades muy extrañas, tiene la capacidad de correr grandes distancias a una velocidad impresionante (utilizando un estilo bastante particular), es inteligente como ningún otro y con su mirada puede percibir casi cualquier cosa. Muchos creen que tiene poderes sobrenaturales ya que se le ha visto controlar aspersores de agua utilizando el poder de su mente, así como ofrendar libros hacia el cielo como haciendo alguna especie de tributo a algún Dios supremo.

Un día un profesor notó todas estas habilidades y decidió hacerle un pequeño desafío. Suponga que el alfabeto consta únicamente de vocales, y para escribir palabras debe considerar las siguientes restricciones: Luego de una 'a' sólo puede colocar 'i' ó 'o', después de una 'e' puede seguir solamente 'a' ó 'u', la 'i' sólo puede ser seguida por 'e', la 'o' por 'i' ó 'u' y por último la 'u' por 'e'. Por ejemplo, "aoiea" es una palabra válida mientras que "aoiau" no porque la letra 'a' no puede ir después de una 'i'. Luego de comprender todo esto, se le pide al pequeño Flash que diga cuál es el número de palabras que pueden ser formadas con este patrón y que su longitud sea igual a N .

Entrada

La entrada consiste en una serie de casos, cada uno incluye un entero N ($1 \leq N \leq 10000$) por línea, y es finalizada por EOF.

Salida

Un simple entero que señale el número de palabras de tamaño N que pueden ser formadas utilizando el patrón descrito anteriormente.

NOTA: el número esperado puede ser un poco grande, así que para evitar inconvenientes imprima el número módulo 1234567.

Ejemplo de entrada

1	1
2	2
3	3
4	

Ejemplo de Salida

1	5
2	8
3	12
4	

Problema G

Googleando Empleos

Nombre de archivo: `goo.c`, `goo.cpp` o `goo.java`
Debe leer desde entrada estándar e imprimir a salida estándar.

LinkedIn es una importante red social para la búsqueda de empleo y de trabajadores, pero tiene una forma peculiar de trabajar, para que una persona A aparezca en la búsqueda de una Persona B, A y B deben ser amigos, o amigo de un amigo, o amigo de un amigo de un amigo, no más de eso. Por ejemplo: X es amigo de Y, Y es amigo de W, y W es amigo de Z, haciendo la búsqueda correcta X puede aparecer en los resultados de Z (o viceversa).



La empresa te ha contratado para diseñar un algoritmo donde puedas determinar si una persona A puede aparecer en los resultados de una persona B.

Entrada

La entrada comienza con un número N que indica la cantidad de casos de prueba, cada uno de estos empieza con un número K , seguido de K líneas compuestas por dos números enteros m, n que representan que la persona m está conectada directamente con la persona n . Seguido a eso un número Q que indica la cantidad de consultas que hará LinkedIn, seguido de Q líneas con dos números enteros J y K que representa una consulta de si J puede aparecer en la búsqueda de K .

Salida

Por cada consulta imprima en una línea True si se da el criterio necesario para aparecer en los resultados o False si no puede aparecer en la red.

Ejemplo de entrada

```
1 1
2 6
3 2 3
4 3 4
5 5 7
6 5 6
7 7 8
8 8 9
9 2
10 2 5
11 5 9
12
```

Ejemplo de Salida

```
1 False
2 True
3
```




Problema H

Hola Mundo

Nombre de archivo: holamundo.c, holamundo.cpp o holamundo.java
Debe leer desde entrada estándar e imprimir a 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 ($1 \leq n \leq 100$). Luego del último caso sigue una línea con 0.

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

```
1 3
2 2
3 0
```

Ejemplo de Salida

```
1 Hola Mundo
2 Hola Mundo
3 Hola Mundo
4
5 Hola Mundo
6 Hola Mundo
7
8
```



Problem I

In the Court of the Crimson King

Filename: crimson.c, crimson.cpp or crimson.java

You must read from standard input and print to standard output.

Let's talk about Robert; he and his friends have been faced by an act from the King Crimson itself. As you may know, the king is a very powerful man, he owns every land you may think of and surely you don't want to have a problem with him.

The king has ordered to Robert to offer their service to the realm, he wants to be able to check how much area there is between two rectangles. (Is a king, he has the right to be capricious)

You decided to help Robert, your old friend, and solve this problem, so the king doesn't get the feel of cut your heads, just saying.

Input

There are several cases, each one in a line with 8 integers, x_1 , y_1 , x_2 , y_2 , x_3 , y_3 , x_4 and y_4 , where (x_1, y_1) and (x_2, y_2) are the coordinates for the first rectangle, and (x_3, y_3) with (x_4, y_4) are the coordinates for the second rectangle. The end of input is signalled with 8 zeroes. All values are between -100 and 100, inclusive.

Output

For each case print one line with the area of the intersection of the two rectangles.

Input sample

```
1 2 2 4 4 3 3 6 6
2 0 0 0 0 0 0 0 0
3
```

Output sample

```
1 1
2
```