

## Problema A - Annie la Hija de la Oscuridad

Límite de tiempo: 3 segundos

### Problema

League of Legends es un videojuego de género MOBA, desarrollado por Riot Games para Microsoft Windows y OS X.

Los jugadores (invocadores), se agrupan en 2 equipos de campeones (champions, o bien champs), 3 vs 3 o 5 vs 5. A partir de diciembre del 2013 hay 118 campeones disponibles en los servidores normales, pero este número aumenta periódicamente. Cada equipo comienza en lados opuestos de un mapa en un área llamada Base, cerca de lo que se llama Nexo (o Nexus). El objetivo del juego es destruir el Nexo del equipo rival. Para destruir un nexo, cada equipo debe llegar a la base enemiga eliminando unas torretas que la protegen.

Cada jugador gana niveles al matar súbditos (NPCs que aparecen constantemente y atacan al otro equipo) del equipo contrario y derrotar a monstruos neutrales. Matar monstruos, campeones enemigos y destruir torretas proporciona oro necesario para mejorar al Campeón y facilitar así las batallas.

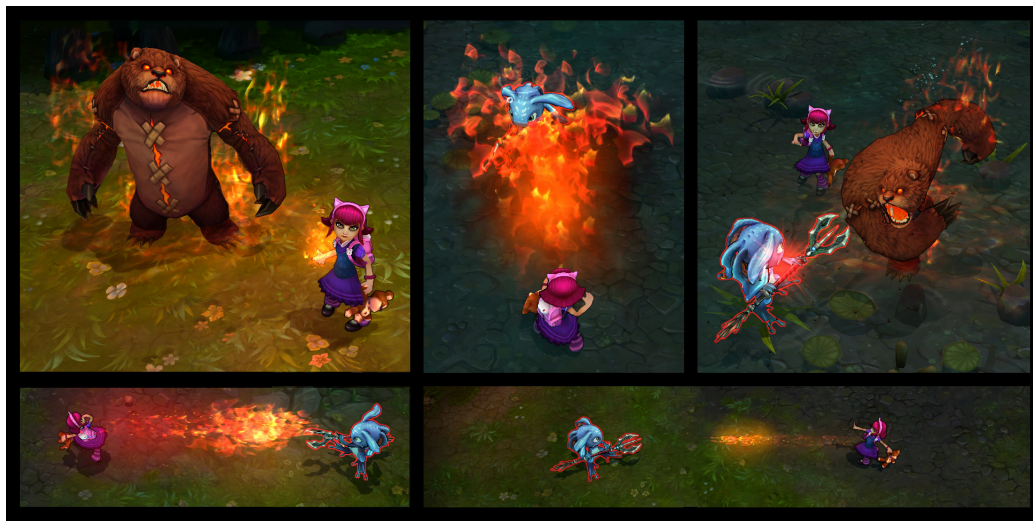


Figura 1: Arriba a la izquierda “Tibbers”, arriba al centro “Incinerar”, Abajo a la izquierda “Desintegrar”

Uno de estos champs es Annie, tu objetivo es ayudarla a determinar cuando podrá encadenar sus tres hechizos ofensivos a un campeón enemigo. Los hechizos que conoce Annie son los siguientes:

- “Desintegrar” Annie lanza una bola de fuego imbuida de Maná a cualquier campeón en un área de  $600u^1$  a su alrededor.

<sup>1</sup> $u$  son unidades de longitud en los campos de la justicia

- “Incinerar” Annie lanza un abrasador cono de fuego, dañando a todos los enemigos de la zona. Este cono tiene una apertura de  $35^\circ$  y además tiene un alcance de  $500u$  (imagine una rebanada de pizza circular, con centro en Annie).
- “Invocar: Tibbers” Tibbers aparece en un estallido de llamas e inflige daño mágico a los enemigos en la zona objetivo. Annie puede invocar Tibbers en cualquier lugar a no más de  $700u$ , Tibbers inflige daño de area donde aparece en un círculo de radio  $75u$ .

Puedes asumir de manera segura que no hay obstáculos entre Annie y su objetivo; además puedes despreciar el tiempo entre el lanzamiento de cada hechizo.

## Entrada

La primer línea de la entrada será un número  $2 \leq T \leq 20$  que será el número de casos. Cada caso comenzará con tres números  $0 \leq A_x, A_y \leq 10000$  y  $1 \leq Q \leq 50$  donde los primeros dos son las coordenadas de Annie y el tercero el número de consultas para ese caso. Las siguientes  $Q$  líneas tendrán dos números  $0 \leq E_x, E_y \leq 10000$  que es la posición de el campeón objetivo. Se asegura que todos los números en la entrada son enteros.

## Salida

Para cada consulta deberas imprimir una línea que dirá “FULL COMBO” si puedes encadenar los 3 hechizos contra el campeón objetivo al mismo tiempo, si no puede, imprime “OUTPLAYED”.

## Entrada Ejemplo

```
2
0 1 1
490 2
23 4 1
700 900
```

## Salida Ejemplo

```
FULL COMBO
OUTPLAYED
```

## Sugerencias

$$c^2 = a^2 + b^2$$

---

Edgar García Rodríguez-Grupo de Algoritmia Avanzada y Programación Competitiva



Facultad de Estudios Superiores

# Acatlán

Centro de Desarrollo Tecnológico  
Departamento de Servicios de Cómputo



## Problema B - Colores

Límite de tiempo: 1 segundo

### Problema

Manuel Nicolás tiene éstos códigos de colores:

0 0 0 = Blanco

0 0 1 = Azul

0 1 0 = Rojo

0 1 1 = Morado

1 0 0 = Amarillo

1 0 1 = Verde

1 1 0 = Naranja

1 1 1 = Negro

Pero su amigo Mir Ist Kalt cambió los códigos, tal que los 1's los volvió 0's, y viceversa.

Tu trabajo es ayudar a Manuel Nicolás a decirle a qué color corresponde cada código cambiado.

### Entrada

La primera línea contendrá un entero  $N$  ( $1 \leq N \leq 20$ ), el número de casos de prueba. Las siguientes  $N$  líneas tendrán 3 dígitos,  $d_{i1}, d_{i2}, d_{i3}$  separados por un espacio ( $0 \leq d_{i1}, d_{i2}, d_{i3} \leq 1$ ) ( $1 \leq i \leq N$ ).

## Salida

Se tendrán que imprimir  $N$  líneas, una por cada caso de prueba, todas terminando con salto de línea; cada línea tendrá el color correspondiente a su código.

## Entrada Ejemplo

```
4
0 0 0
1 1 1
0 0 1
0 1 1
```

## Salida Ejemplo

```
Negro
Blanco
Naranja
Amarillo
```

---

Sergio Adrián Lagunas Pinacho - Grupo de Algoritmia Avanzada y Programación Competitiva



## Problema C - CandyLand

Límite de tiempo: 1 segundo

### Problema

Mágicamente has sido transportado a la asombrosa tierra de CandyLand, donde hay tantos dulces de tantos sabores como nunca habías imaginado. Siendo un fanático de los dulces no puedes contener la felicidad y empiezas a probar todos los dulces existentes y asignas a cada dulce un valor de sabrosidad. Sin embargo no todo puede ser tan perfecto, tu tiempo en CandyLand está a punto de acabar y serás transportado de regreso a tu vida cotidiana, pero no quieres irte con las manos vacías.

Enfrente de ti hay una línea con  $1 \leq N \leq 10^6$  dulces numerados del 1 al  $N$ , donde al  $i$ -ésimo dulce le has asignado un valor de sabrosidad  $-100 \leq S_i \leq 100$ . Y convenientemente tienes a tu disposición un brazo robótico capaz de agarrar exactamente  $1 \leq K \leq 10^3, K \leq N$  dulces consecutivos de dicha línea, y una computadora. Debido a que hay algunos dulces que no te gustan tanto, quieres agarrar  $K$  dulces tales que la suma de su valor de sabrosidad sea el máximo posible. Como la cantidad de dulces es muy grande necesitas hacer un programa que te diga cuál es el índice del primer dulce de la izquierda que debe agarrar el brazo robótico para lograr tu objetivo.

### Entrada

La primer línea contendrá el número  $1 \leq T \leq 50$  de casos. Para cada caso habrá dos líneas. La primer línea de cada caso contendrá dos enteros  $N$  y  $K$ . La segunda línea de cada caso contendrá  $N$  enteros  $S_i, 1 \leq i \leq N$ .

### Salida

Para cada caso deberás imprimir dos enteros en una línea, el índice del primer dulce de la izquierda que debe agarrar el brazo robótico para lograr tu objetivo, si hay más de una opción imprime el que tenga el menor índice, y la suma máxima del valor de sabrosidad que puedes conseguir.

### Nota

El brazo robótico tiene que agarrar exactamente  $K$  dulces.

### Entrada ejemplo

```
1
10 4
1 -5 5 10 -1 3 -2 -3 9 4
```

### Salida Ejemplo

```
3 17
```



Facultad de Estudios Superiores

# Acatlán

Centro de Desarrollo Tecnológico  
Departamento de Servicios de Cómputo



## Problema D - Raíces digitales

Límite de tiempo: 1 segundo

### Problema

Recientemente Manuel Nicolás León descubrió qué es la raíz digital, y decidió compartir su conocimiento contigo.

Digamos que  $S(n)$  es la suma de los dígitos de  $n$ , por ejemplo,  $S(4089) = 4 + 0 + 9 + 8 = 21$ , entonces la raíz digital del número  $n$  es:

1.  $rd(n) = S(n)$  si  $S(n) < 10$
2.  $rd(n) = rd(S(n))$  si  $S(n) \geq 10$

Por ejemplo,  $rd(4098) = rd(21) = rd(3) = 3$ .

Manuel le tiene miedo a los números grandes, por eso los números con los que trabajará serán a lo más  $10^{100}$ . Para todos esos números, Manuel ha probado que  $rd(n) = S(S(S(S(n))))$  ( $n \leq 10^{100}$ ).

Ahora Manuel quiere encontrar números rápidamente dada su raíz digital. El problema es que todavía no ha aprendido a hacer lo que te va a preguntar. Tu tarea es, dados los números  $k$  y  $d$ , encontrar números exactamente de  $k$  dígitos (sin ceros al principio), con su raíz digital igual a  $d$ .

### Entrada

La primera línea tendrá un número  $T$  ( $T \leq 500$ ) que representa el número de casos de prueba.

Las siguientes  $T$  líneas contendrán los casos de prueba, cada caso tendrá 2 números,  $k$  y  $d$  ( $1 \leq k \leq 100$ ;  $1 \leq d \leq 9$ ).

### Salida

Se imprimirán  $2T$  líneas, 2 por cada caso de prueba: la primera línea de cada caso de prueba tendrá el número  $n$  más grande tal que  $rd(n) = d$ , y el número de dígitos de  $n$  sea igual a  $k$ ; la segunda línea de cada caso de prueba tendrá el número  $n$  más chico tal que  $rd(n) = d$ , y el número de dígitos de  $n$  sea igual a  $k$ .

Puedes tener la seguridad de que dichos números siempre existen, y son únicos.

El primer dígito de cada número impreso no tiene que ser un 0.

### Entrada Ejemplo

2  
1 3  
1 7

### Salida Ejemplo

3  
3  
7  
7

---

Sergio Adrián Lagunas Pinacho - Grupo de Algoritmia Avanzada y Programación Competitiva



Facultad de Estudios Superiores

# Acatlán

Centro de Desarrollo Tecnológico  
Departamento de Servicios de Cómputo



## E - Buscando asiento

Límite de tiempo: 1 segundo

### Problema

Manuel Nicolás León es un chico muy especial... cuando va a conferencias le gusta que la silla de su izquierda y la silla de su derecha estén desocupadas.

Éste año irá a una conferencia a la cuál irán sólo chicos igual de especiales que él (que les gusta que las sillas adyacentes a la suya estén desocupadas). Para el problema tienes que decir si es posible que todos los chicos, incluyendo Manuel, estén a gusto, ésto es, que todos los chicos tengan libres las sillas adyacentes a la suya.

### Entrada

La primera línea tendrá un número  $T$  que representa el número de casos de prueba.

Las siguientes  $T$  líneas contendrán los casos de prueba, cada caso tendrá 2 números,  $A$  y  $B$  ( $1 \leq A \leq B \leq 500$ ) indicando el número de chicos, y el número de sillas disponibles respectivamente.

Los chicos escogen las sillas uno por uno, escogiendo una silla que tenga sus 2 sillas adyacentes desocupadas (Si es una de las sillas de la orilla, basta con que la única silla adyacente que tiene esté desocupada).

Tu tarea es, dadas éstas condiciones, decir si los chicos se sentirán a gusto o no.

### Salida

Se imprimirán  $T$  líneas, una por cada caso de prueba, con alguna de las siguientes palabras: “Si” si siempre es posible que todos los chicos estén a gusto, “No” si es imposible que todos los chicos estén a gusto, o “Tal vez” si depende de cómo se vayan sentando los chicos.

### Entrada Ejemplo

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

### Salida Ejemplo

```
Si
Tal vez
No
```





Facultad de Estudios Superiores

# Acatlán

Centro de Desarrollo Tecnológico  
Departamento de Servicios de Cómputo



## F Fibonacci

Límite de tiempo: 1 segundo

### Problema

Manuel Nicolás León es un chico muy curioso, tanto que durante una de sus clases se le ocurrió una sucesión de dígitos parecida a la de fibonacci, la cuál empieza con 2 dígitos, 1 y 1, y para la cual cual(es)quiera dígito(s) que se le vaya a añadir es igual a los dígitos que equivalen a la suma de los valores numéricos de los 2 dígitos anteriores, así el tercer dígito sería  $1+1=2$ , el cuarto sería  $1+2=3$ , el quinto  $2+3=5$ , el sexto  $3+5=8$ , y así sucesivamente.

Lo interesante de ésta sucesión empieza del dígito 7 en adelante; dado que  $5+8=13$ , los dígitos 7 y 8 serían el 1 y el 3 respectivamente; el dígito que ocuparía la posición 9 sería  $1+3=4$ , y así sucesivamente, dando lugar a algo como ésto:

112358134...

Manuel Nicolás León se pregunta si hay alguna forma de saber cualquier dígito de la sucesión con ayuda de un programa, para lo cuál te ha pedido tu ayuda: tienes que programarlo.

### Entrada

La primera línea contendrá un entero  $N$  ( $1 \leq N \leq 20$ ), el número de casos de prueba. Las siguientes  $N$  líneas tendrán un número entero positivo  $S_i$  ( $1 \leq S_i \leq 10^9$ ) ( $1 \leq i \leq N$ ) que representa el dígito requerido.

### Salida

Se tendrán que imprimir  $N$  líneas, una por cada caso de prueba, todas terminando con salto de línea; cada línea tendrá un dígito  $D_i$  ( $0 \leq D_i \leq 9$ ) ( $1 \leq i \leq N$ ) que representa el dígito que está en la posición  $S_i$  requerido.

### Entrada Ejemplo

6  
1  
2  
3  
7  
8  
9

### Salida Ejemplo

1  
1  
2  
1  
3  
4

---

Sergio Adrián Lagunas Pinacho - Grupo de Algoritmia Avanzada y Programación Competitiva

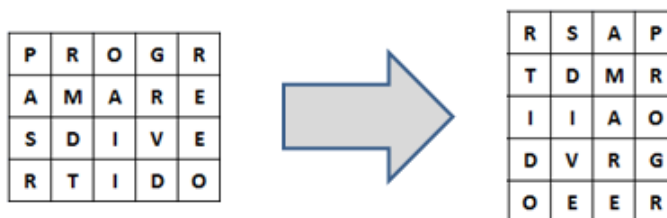
## Problema G - Código

Límite de tiempo: 1 segundo

### Problema

Mau es un chico al que le encanta descifrar códigos. Recientemente un amigo suyo le contó acerca de un nuevo código que ha estado desarrollando. Este código consiste en colocar el mensaje a codificar sin espacios en una cuadrícula de  $M \times N$ , siendo  $M$  el número de renglones y  $N$  el número de columnas. Luego se rota la cuadrícula 90 grados. Así, al leer normalmente el mensaje colocado en la cuadrícula, éste será incomprensible.

Un ejemplo se muestra en la siguiente imagen:



Mau puede decodificar este código, pero no lo suficientemente rápido, por eso te ha pedido ayuda a ti, su amigo programador, para implementar un programa que descifre este tipo de códigos.

Tu tarea será, dado el mensaje codificado, imprimir el mensaje original que fue colocado en la cuadrícula.

### Entrada

La primera línea contiene un entero  $T$  ( $1 \leq T \leq 50$ ), el número de casos de prueba. Para cada uno de los siguientes  $T$  casos siguen 2 líneas.

La primera de ellas contiene dos enteros  $M$  y  $N$ , ( $1 \leq M, N \leq 50$ ), que representan el número de renglones y columnas respectivamente en las que está colocado el mensaje original.

La segunda contiene el mensaje codificado, de tamaño  $M \times N$  el cual podrá estar formado de letras mayúsculas, letras minúsculas y números.

### Salida

Para cada caso imprime solo una línea que debe contener el mensaje original, con la distinción de mayúsculas y minúsculas.

### Entrada Ejemplo

```
1
4 5
RSAPTDMRIIAODVRGOEER
```

### Salida Ejemplo

```
PROGRAMARES DIVERTIDO
```

---

Maximiliano Vera Luna - Grupo de Algoritmia Avanzada y Programación Competitiva