

Cuarta maratón de programación de los cursos de Algoritmos.

Categoría Dijkstra

Instrucciones

1. URL del juez: <https://acm.javeriana.edu.co/maratones/2016/Dijkstra/>

Maratón de Programación Interna de Algoritmos

Es un espacio de sana competencia donde los estudiantes de los cursos de algoritmos trabajan en equipos para resolver la mayor cantidad de ejercicios en el menor tiempo posible.

Los ejercicios propuestos en la maratón combinan pensamiento lógico, habilidades algorítmicas y deben ser resueltos a través de un programa de computador escrito en lenguaje Java bajo unas condiciones especificadas desde el inicio de la competencia.

Cada equipo estará conformado por máximo dos estudiantes quienes deben elegir un nombre que los identifique para registrarse en la maratón. El evento cuenta con tres categorías cada una con su propio conjunto de problemas y orientada a un grupo particular de estudiantes.

La participación de los estudiantes es en equipos de dos personas. Cada equipo debe tener un nombre que lo identifique en la maratón y debe ser definido en el momento de la inscripción. La organización de la competencia entregará premiación a los tres primeros equipos de cada categoría. La participación en la maratón no es obligatorio y, debido a restricciones de espacio, los cupos son limitados.

Reglamento:

1. SEDE Y FECHA

La maratón interna de algoritmos de la universidad Icesi se realiza en la universidad Icesi en las salas de cómputo 202C y 203C. Para el primer semestre de 2016, ésta se realizará el Viernes 13 de Mayo de 2016.

2. CATEGORÍAS

La maratón interna se desarrollará en 3 categorías, en ellas podrán participar estudiantes que esté cursando a lo sumo el curso descrito en la categoría, estas son:

- **Categoría Dijkstra.** Para los estudiantes de Algoritmos y Programación I - Fundamentos de programación en diseño.
- **Categoría Turing.** Para los estudiantes de Algoritmos y Programación II - Diseñando con Algoritmos.
- **Categoría Borg.** Para los estudiantes de Algoritmos y Estructuras de Datos y semestres posteriores.

En cuanto a restricciones de género no hay ninguna, todas las categorías son abiertas por lo que los equipos pueden ser 2 personas del mismo género, o mixto.

3. REGISTRO DE PARTICIPACIÓN

Podrán participar los estudiantes de la universidad Icesi inscribiéndose por medio de un formulario web que se les hará llegar via correo electrónico.

4. MÍNIMOS PARA LA REALIZACIÓN DE LAS COMPETENCIAS

Para que exista competencia el número mínimo deberá ser de cinco (5) equipos en la categoría correspondiente.

5. PREMIACIÓN

En la maratón interna de Icesi se estarán disputando, 3 Medallas de oro, 3 medallas de plata y 3 medallas de bronce. Las cuales serán distribuidas 1 por categoría.

La premiación oficial para la maratón interna de algoritmos es la siguiente:

1er Puesto: Medalla Dorada, Bono Ventolini, Matrícula Preferencial

2do Puesto: Medalla Plateada, Bono Cafetería Icesi

3er Puesto: Medalla de Bronce.

6. CONDICIONES PARA SER ACREEDOR A LA PREMIACIÓN

La “Matrícula Preferencial” es la ayuda que le brinda la universidad Icesi a las personas que la representan en competencias, la cual consiste en dar prioridad durante la semana de matrículas a los deportistas, para que de esta manera puedan matricular los cursos de su semestre sin afectar sus horarios de entreno. Así es como se incentivan a los deportistas a seguir participando por la universidad en los diferentes torneos que se realizan durante la jornada académica.

Ya que este reconocimiento es uno de los más grandes que se puede dar a nivel académico, para que el estudiante sea acreedor de ella debe cumplir las siguientes condiciones:

- A. Obtener el primer puesto en la categoría inscrita.
- B. Su equipo debe haber resuelto al menos un problema en la maratón.
- C. No puede ser repitente del curso
- D. No puede haber cancelado el curso en semestres anteriores.
- E. No puede estar en prueba académica.
- F. No puede tener procesos disciplinarios pendientes.
- G. Debe asistir el próximo semestre al club de programación.

Tabla de Contenido

- Problema A: Duelo de magos
- Problema B: Las esferas del dragon
- Problema C: Profesor Perezoso
- Problema D: Secuencias
- Problema E: Sumas Binaria
- Problema F: Piedra, papel, tijera, lagarto, spock

Problema A: Duelo de Magos

Base Name: duelo.java

Autor: Camilo Barrios - Universidad Icesi

Harry Potter y El-Que-No-Debe-Ser-Nombrado están en una lucha a muerte una vez más. Esta vez se encuentran frente a frente, en los extremos opuestos del pasillo de longitud L en Hogwarts (Si no sabe quienes son los personajes mencionados, hay que leer). Los oponentes cargan simultáneamente un hechizo mortal (*Avada Kedavra*) y lo lanzan hacia su enemigo. Sabemos que el impulso del hechizo mágico de Harry vuela a una velocidad de P metros por segundo, y el impulso de hechizo mágico de El-Que-No-Debe-Ser-Nombrado vuela a una velocidad de Q metros por segundo.

Los impulsos mágicos se mueven a través del pasillo uno hacia el otro. En el momento que colisionan ellos “rebotan” y cambian de dirección; ellos viajan de nuevo hacia la persona que los lanzó con la misma velocidad inicial con que fueron lanzados. Entonces, tan pronto como el impulso llega de nuevo a su lanzador, el lo refleja y la envía de nuevo hacia el enemigo, sin cambiar la velocidad original del impulso.

Como todos sabemos Harry ha dominado perfectamente los fundamentos de la magia, él sabe que después de la segunda colisión ambos impulsos van a desaparecer, y una potente explosión ocurrirá exactamente en el lugar de la colisión. Sin embargo, el joven mago no es bueno para las matemáticas, por lo que le pide calcular la distancia desde su posición hasta el lugar de la segunda colisión de los impulsos de hechizos, sabiendo que los oponentes no cambian de posición durante toda la pelea.

Entrada

La primera línea sera un numero N que dirá cuántos casos de prueba vienen a continuación, puedes asumir que $N \leq 500$. Cada caso de prueba consiste en dos líneas, la primera línea contiene un entero L , el cual representa la distancia en metros del corredor en donde se presenta la pelea, puedes asumir que ($1 \leq L \leq 1000$). La segunda línea contiene 2 números enteros, P y Q respectivamente, estos números vendrán separados por un espacio y representan las velocidades de los impulsos mágicos de Harry y El-Que-No-Debe-Ser-Nombrado. Podemos asumir que ($1 \leq p, q \leq 500$).

Salida

Para cada caso de prueba, la salida de una línea que contiene "Duelo #x: y", donde x es el número del duelo (empieza a partir de 1) mientras que y es la distancia que existe entre la posición original de Harry y el lugar donde se produce la explosión (El lugar donde se colisionan por segunda vez ambos impulsos mágicos). Es necesario que la respuesta tenga una cifra decimal en todos los casos.

Ejemplo

Entrada	Salida
4 100 50 50 199 60 40 771 33 17 99 123 84	Duelo #1: 50,0 Duelo #2: 119,4 Duelo #3: 508,9 Duelo #4: 58,8

Problema B: Las Esferas del Dragón

Base Name: `esferas.java`

Autor: Juan Manuel Reyes - Universidad Icesi

Las esferas del dragón son varias esferas mágicas de cristal que están dispersas y cuando se reúnen son utilizadas para invocar al dragón Shenlong que concede uno o varios deseos. Esta invocación hace que el cielo del planeta donde se activen, se nuble y oscurezca. Su color siempre es naranja cristalino, salvo cuando están cargadas de energía negativa, y están marcadas con estrellas en relieve en su interior. Las esferas varían de 1 a n dependiendo del planeta donde se encuentren. Una vez concedido el deseo o deseos, se convierten en simples piedras durante el plazo de un año y se dispersan por todo el planeta nuevamente.

Son Gokú es un niño aún y necesita encontrar las esferas del dragón, pero solo tiene un dispositivo fabricado por la Corporación Cápsula que le entrega unas coordenadas de ubicación de las esferas pero él no puede entender que significan esos números. Lo que necesita Goku es poder transformar los datos arrojados por el dispositivo en un mapa legible que le permita encontrar las esferas para reunirlos y poder pedir el deseo de revivir a su abuelo.

Entrada

La primera línea contiene un número entero $0 < c \leq 100$ con la cantidad de casos de prueba. Luego siguen c casos de prueba así: la primera línea de cada caso tiene tres números enteros $1 \leq h \leq 50$, $1 \leq w \leq 50$, $0 \leq n \leq h * w$. Los dos primeros son las dimensiones de la superficie del planeta donde se ubican las esferas y el tercero es la cantidad de esferas de ese planeta. Luego, en ese caso vienen n líneas con un par de números enteros $0 \leq i < h$, $0 \leq j < w$ indicando la posición, a través de coordenadas, de las n esferas en la superficie.

Salida

Para cada caso de prueba, un mapa dibujado con símbolos `+`, de la superficie incluyendo las esferas indicadas en las coordenadas, con el símbolo `O` (letra o mayúscula). Después de cada mapa hay un salto de línea.

Ejemplo

Entrada	Salida
3	+++++++
5 4 7	+ +0+ + +
2 3	+++++++
1 2	+ + +0+ +
3 0	+++++++
4 2	+0+ + +0+
0 1	+++++++
2 0	+0+ + + +
4 1	+++++++
2 3 3	+ +0+0+ +
0 0	+++++++

1 1
0 2
3 6 5
0 5
1 4
2 3
1 2
0 3

```
+++++++
+0+ +0+
+++++++
+ +0+ +
+++++++

+++++++
+ + + +0+ +0+
+++++++
+ + +0+ +0+ +
+++++++
+ + + +0+ + +
+++++++
```

Problema C: Profesor Perezoso

basename: perezoso.java

Un profesor muy perezoso de la Universidad le ha pedido ayuda a usted para obtener las calificaciones del examen final de su materia (Departamento del Matemáticas). El profesor le dará la cantidad de preguntas N , y acto seguido le dará N parejas de números (a, b) , a siendo la respuesta correcta y b siendo la respuesta del estudiante. Como el profesor es tan perezoso no le importa el procedimiento, solo si la respuesta del estudiante es igual a la respuesta correcta. El profesor es tan perezoso, que no quiere escribir ningún decimal después de la coma (,) es decir, si su calificación fue 4,6 el profesor le pondrá un 4. Usted debe imprimir la nota sobre 5 en cada uno de los casos de prueba.

Entrada

La primera línea de la entrada es un numero $T < 100$, indicando la cantidad de casos de prueba. Por cada T , recibirá un numero $1 \leq N \leq 20$ que indican la cantidad de preguntas de ese caso de prueba. Seguirán N línea cada una con una pareja de números $-100 \leq a, b \leq 100$, estos dos números estarán separados por un espacio.

Salida

Usted deberá imprimir T líneas, cada una con la calificación sobre 5 y sin números decimales, para cada examen de N preguntas.

Ejemplo

Entrada	Salida
2	3
5	2
1 1	
2 3	
4 4	
-100 100	
5 5	
6	
1 1	
2 2	
3 3	

4 5 6 7 100 99	
----------------------	--

Problema D: Secuencias

Base Name: `secuencia.java`

Autor: Juan Sebastián Prada

Andrés es un joven al cual le gustan mucho las matemáticas y en especial las secuencias, una secuencia es una serie de números a los cuales se les ha aplicado una serie de operaciones.

Por ejemplo si tengo la serie de operaciones +5 -3 e inicio en el número 4 tendría una secuencia como esta:

4 9 6 11 8 13 10...

Para este ejercicio Andrés le ha pedido a usted que dado un número después de una ronda completa de operaciones de una secuencia cualquiera usted encuentre el menor número mayor a 0 con el cual inicio.

Entrada

Para cada caso habrá dos líneas en la primera se tendrá el numero con el que termino la secuencia después de una serie de iteraciones. En la segunda línea se tendrá la serie de operaciones con la cual se obtuvo ese número.

Salida

Para cada caso imprimir el menor número mayor a 0 con el cual se inició la secuencia, en caso de no existir un número mayor a 0 con el cual haya iniciado la secuencia imprimir imposible.

Ejemplo

Entrada	Salida
37 6+ 9- 5+ 5- 9+ 9 8+ 8- 31	7 9 3

7+	
----	--

Problema E: Binarios

Nombre del Archivo: binarios.java

Autor: Felipe Clement

Un número binario nace del sistema de numeración binario, el cual consiste de solo ceros y unos. Estos ceros y unos se multiplican por potencias de 2 para encontrar su representación decimal. Ejemplo:

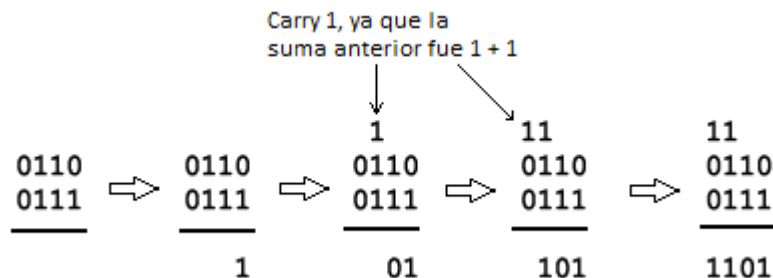
$$01010 \text{ en decimal es: } (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) = 0 + 8 + 0 + 2 + 0 = 10$$

A los números binarios se les puede aplicar también las operaciones de suma, resta, multiplicación, división, etc.

La suma de dos números binarios se hace bit por bit, empezando por la derecha. Donde:

- $0 + 0 = 0$
- $0 + 1 = 1$
- $1 + 0 = 1$
- $1 + 1 = 0$ (carry 1)

El carry significa que se le sumara un 1 a la siguiente suma. Un ejemplo de una suma binaria es:



Dados dos números binarios, usted debe realizar la suma.

Entrada

La entrada contiene varios casos de prueba, cada caso de prueba consiste de dos líneas. Estas dos líneas tienen cada una un número binario, el primer número es de longitud M ($M \leq 10^4$) y el segundo número binario de longitud N ($N \leq M$).

Salida

Por cada dos números binarios, usted debe imprimir su suma.

Ejemplo

Entrada	Salida
0110	1101
0111	11111111
10101010	000111
1010101	10
000000	
111	
1	
1	

Problema F: Piedra, papel, tijera, lagarto y spock

Base Name: spock.java

Autor: Brayan Henao

Piedra, papel, tijera, lagarto y spock es una variación del popular juego piedra papel y tijera. Ambos operan bajo el mismo principio, solo que se han adicionado 2 nuevas armas: el lagarto (formado por la mano como la boca de un lagarto) y spock (formado por el saludo de los Vulcans en Star Trek). Esto hace que se reduzcan posibilidades de que la ronda quede empatada. Este juego fue inventado por Sam Kass y Karen Bryla. En esta versión, los posibles resultados de un encuentro son los siguientes:



- Las tijeras cortan el papel
- El papel cubre la piedra
- La piedra aplasta al lagarto
- El lagarto envenena a spock
- Spock destroza las tijeras
- Las tijeras decapitan al lagarto
- El lagarto se come el papel
- El papel refuta a spock
- Spock vaporiza la piedra
- La piedra aplasta las tijeras

Un día, dos amigos, Rajesh y Sheldon deciden apostar quien pagará el almuerzo del otro, para esto deciden utilizar este entretenido juego. Su misión es escribir un algoritmo que, basado en los un escogió cada uno, decir quién es el ganador o si hay un empate.

Entrada

Habrán varios casos de prueba. La primera línea contiene un entero T, el número de casos de Prueba. Siguiendo habrá T líneas, cada una correspondiente a un caso de prueba. Cada caso de prueba contendrá 2 palabras, el arma elegida por Rajesh y el arma elegida por Sheldon respectivamente.

Salida

Para cada caso de prueba, se debe imprimir "Bazinga" si el ganador es Sheldon, "Rajesh" si el ganador es Rajesh o "Empate" si hay un empate entre ambos participantes

Ejemplo

Entrada	Salida
4 papel tijera spock spock tijeras spock lagarto spock	Bazinga Empate Bazinga Rajesh