



Concurso de Programación - Entrenamiento AdaByron 2020

<http://www.ada-byron.es>

Cuadernillo de problemas

Patrocinado por


accenture

ThoughtWorks®



Realizado en la **Escuela Técnica Superior de Ingeniería Informática (URJC)**
28-29 de febrero de 2020

 Universidad
Rey Juan Carlos

In almost every computation a great variety of arrangements for the succession of the processes is possible, and various considerations must influence the selections amongst them for the purposes of a calculating engine. One essential object is to choose that arrangement which shall tend to reduce to a minimum the time necessary for completing the calculation.

Ada Byron

Listado de problemas

A Primavera: Hola Abril	3
B Verano: ¿Cuántos pañales me llevo de vacaciones?	5
C Otoño: Recogiendo Hojas	7
D Invierno: Contando días	9

Autores de los problemas:

- Isaac Lozano Osorio (Universidad Rey Juan Carlos)
- Jakub Jan Luczyn (Universidad Rey Juan Carlos)
- Jesús Sanchez-Oro Calvo (Universidad Rey Juan Carlos)



Primavera: Hola Abril

Durante el próximo mes tendremos un nuevo miembro más en la programación competitiva llamada Abril, la más joven de todos. Esto es debido a que uno de los profes será padre, por eso mismo en este problema queremos darle la bienvenida. Este problema no tiene ninguna entrada, solo se le debe dar la bienvenida imprimiendo “Hola Abril”

Entrada

Este programa no tiene entrada.

Salida

Salida esperada “Hola Abril”.

Entrada de ejemplo

Salida de ejemplo

Límites

- Ninguno.
- Tiempo: 1 segundo.

● B

Verano: ¿Cuántos pañales me llevo de vacaciones?

Horse Luis está planificando las vacaciones de verano y tiene un grave problema. Con la llegada del nuevo miembro de la familia, Horse Lito, tiene que organizar el número de pañales que necesita llevarse para que no haya problemas durante las vacaciones. La familia al completo, compuesta por Horse Luis, su mujer Horse Fina y el pequeño Horse Lito se van una semana a un destino sin tiendas de pañales. Por esto, necesita hacer una estimación de los pañales que necesitará en función del número de pañales que ha ido gastando en las últimas semanas. El número de pañales gastado depende del estado de Horse Lito en cada momento (por ejemplo, si ha estado enfermo ha gastado más). Horse Luis ha sido previsor y sabe los pañales gastados en cada estado en las últimas semanas. Sabiendo esto, dado un estado concreto, necesita saber cuál será el número de pañales que necesita, que será el máximo de ese estado más uno, para evitar escapes inesperados.

Entrada

La primera línea contiene un entero S que representa el número de semanas que va a analizar. Por cada semana aparecerá un entero E que indica el número de estados de esa semana, y por cada estado una línea con una cadena C y un entero N que indican el nombre del estado y el número de pañales gastados, respectivamente. Finalmente aparecerá un entero P con el número de casos de prueba y a continuación P líneas con el nombre de un estado C en cada una de ellas.

Salida

Para cada caso de prueba se deberá imprimir por consola el máximo de pañales necesario para ese estado, o el carácter '?' si no se conoce nada de ese estado.

Entrada de ejemplo

```
2
3
Enfermo 10
Normal 20
Colegio 25
4
Colegio 10
Enfermo 50
Abuelos 40
Dormir 5
3
Enfermo
Normal
Deporte
```

Salida de ejemplo

```
51
21
?
```

Límites

- $1 \leq E, S \leq 1000$
- $1 \leq C \leq 30$ letras
- $1 \leq N \leq 100$
- $0 \leq P \leq 250000$
- Tiempo 0.5s



Otoño: Recogiendo Hojas

Durante la estación de otoño la temperatura comienza a descender. Las hojas de los árboles caducos cambian su color verde por tonos ocres, hasta que se secan y caen ayudadas por el viento que sopla con mayor fuerza. Este cambio de color se observa claramente en diversas regiones del mundo, incluido Autumnus, una ciudad ficticia creada para satisfacer las necesidades de este problema.

El alcalde de Autumnus, Vertumno, ha conseguido predecir, gracias a varios informes, que este otoño van a caer tantas hojas que las zonas designadas para tirarlas podrían desbordarse. Preocupado por tal posibilidad, Vertumno ha decidido poner en marcha un plan de emergencia para deshacerse de las hojas recogidas.

El plan en cuestión consiste en construir un incinerador de grandes dimensiones para quemar semanalmente las hojas que se han ido acumulando en la ciudad. Ahora solamente queda el problema de logística, transportar las hojas. Se ha decidido designar una flota de camiones para esta tarea. Cada camión aparcará en el lugar de las calles de la ciudad más cercano (incluyendo las intersecciones) al punto de recogida; el tiempo que se tarda en recorrer un trozo de una calle es directamente proporcional a la longitud recorrida. A continuación, los operarios transportarán las hojas hasta el camión con una carretilla, que avanza una unidad de distancia en diez unidades de tiempo.

El alcalde quiere que la recogida se complete en el menor tiempo posible. Por lo que necesitará ayuda para planificar todas las rutas.

Entrada

La primera línea contiene un número N , el número de intersecciones en la ciudad, C el número total de calles que conectan dichas intersecciones, y R , el número de puntos de los que hay que recoger hojas. Las N líneas siguientes indicarán las coordenadas enteras (X, Y) de las intersecciones. Después C líneas indicando, O y D , los dos extremos de la calle, siempre bidireccional, y T , el tiempo que se tarda en recorrerla, incluyendo las estimaciones del tráfico. Las últimas C líneas contendrán las coordenadas enteras (X, Y) de los puntos de recogida de hojas.

El incinerador estará siempre colocado en la primera intersección y todas las coordenadas serán únicas. Además, en Autumnus, se puede llegar desde una intersección a cualquier otra y no has dos calles conectando el mismo par de intersecciones.

Salida

Se debe imprimir el tiempo total, con seis cifras decimales redondeadas, que tardaría un único camión situado en el incinerador en traer todas las hojas; teniendo en cuenta también el tiempo que tardan los operarios en hacer el viaje de ida y vuelta en carretilla con las hojas, suponiendo un único viaje para transportarlas. Cada cargamento debe depositarse en el incinerador antes de recoger otro.

Entrada de ejemplo

```
5 5 2
4 9
0 9
10 4
3 9
1 8
2 1 5
3 1 6
5 1 14
4 1 17
3 2 13
10 6
4 7
```

Salida de ejemplo

61.161638

Límites

- $1 \leq N \leq 100000$
- $1 \leq C \leq 500000$
- $1 \leq R \leq 10$
- $1 \leq O, D \leq N$
- $1 \leq T \leq 50000$
- $-5000 \leq X, Y \leq 5000$
- Tiempo: 2.25 segundos.



Invierno: Contando días

No hace mucho tiempo que fue el SWERC (Southwestern Europe Regional Contest), al cual la URJC llevó tres equipos. Algunos seguro que están deseando entrenar para poder ir. Normalmente, estos equipos hacen planes de entrenamiento, ejercicios, no saldrán de fiesta, y apenas dormirán, ya que el objetivo principal será ganar el SWERC. Sin embargo, cualquier tipo de planificación parte de la identificación del número de días que quedan para el evento en cuestión. Por ello, se necesita desarrollar un programa que calcule automáticamente cuántos días quedan para el siguiente SWERC.

Entrada

La primera línea contiene un número T , que especifica el número de casos de prueba. Seguidamente vendrán $2T$ líneas que denotan los diferentes casos de prueba. Cada caso de prueba está compuesto de dos líneas donde cada una denota una fecha. La primera fecha será la fecha desde la cual se quiere iniciar el cálculo y la segunda línea contendrá la fecha de celebración del SWERC. Cada una de ellas viene representada con 3 números D día, M mes, A año. La segunda fecha siempre es superior o igual a la primera.

Salida

Dadas dos fechas, teniendo en cuenta años bisiestos. ¿Cuántos días faltan para el SWERC?.

Entrada de ejemplo

```
3
10 2 2014
10 3 2015
10 2 2000
10 3 2000
1 2 2000
1 2 2004
```

Salida de ejemplo

```
393
29
1461
```

Límites

- $1 \leq T \leq 10000$
- $1 \leq D \leq$ Máximo día por mes
- $1 \leq M \leq 12$
- $0 \leq A \leq 5000$
- Tiempo 0.5s